*open*FT V10.0 for BS2000/OSD

Installation and Administration

System Administrator Guide

Edition May 2008

Comments… Suggestions… Corrections…

The User Documentation Department would like to know your opinion on this manual. Your feedback helps us to optimize our documentation to suit your individual needs.

Feel free to send us your comments by e-mail to manuals@fujitsu-siemens.com.

Certified documentation according to DIN EN ISO 9001:2000

To ensure a consistently high quality standard and user-friendliness, this documentation was created to meet the regulations of a quality management system which complies with the requirements of the standard DIN EN ISO 9001:2000.

cognitas. Gesellschaft für Technik-Dokumentation mbH
www.cognitas.de

Copyright and Trademarks

Copyright © Fujitsu Siemens Computers GmbH 2008.

All rights reserved.
Delivery subject to availability; right of technical modifications reserved.

All hardware and software names used are trademarks of their respective manufacturers.

Contents

1 Introduction 9

1.1 Brief description of the product *open*FT 10
1.2 Target group and objectives of this manual 10
1.3 Concept of *open*FT for BS2000/OSD manuals 11
1.4 Organization of the System Administrator Guide 12
1.5 Changes since the last version of the manual 13
1.6 README file 14

2 Installation and startup 15

2.1 Installing *open*FT 15
2.1.1 Initial installation of *open*FT for BS2000/OSD 16
2.1.2 Version change and compatibility 17
2.1.3 Installation of the command interface for POSIX 17

2.2 Startup 18
2.2.1 Preparing the FT system 18
2.2.2 Entering partners in the partner list 20
2.2.3 Starting and stopping *open*FT 21

2.3 Installing *open*FT-AC 22
2.3.1 Initial installation 22
2.3.2 Version change 22

2.4 Installing the graphical interfaces 23
2.4.1 Connection setup 24
2.4.2 Configuration 25

2.5 Configuring *open*FT-AC 26

3 Operation 29

3.1 Optimizing the operating parameters 31
3.1.1 Interdependencies for optimized parameterization 31
3.1.2 Achieving optimized operation 32
3.1.3 Changing the PROCESS-LIMIT operating parameter 33
3.1.4 Changing the CONNECTION-LIMIT operating parameter 34
3.1.5 Changing the TRANSPORT-UNIT-SIZE operating parameter 35
3.1.6 Setting the MAX-REQUEST-LIFETIME operating parameter 35

3.2 Administering code tables 36

3.3 Administering requests 37

3.4 Administering partners 38
3.4.1 Defining partner properties 38
3.4.1.1 Specifying partner addresses 39
3.4.1.2 FTAC security levels for partner entries 42
3.4.1.3 Automatic deactivation 42
3.4.2 Dynamic partners 43
3.4.3 Backing up the partner list 43
3.4.4 Addressing concept for partners up to *open*FT V8.0 44

3.5 Security in FT operation 46
3.5.1 Authentication 46
3.5.2 Extended authentication check 51
3.5.3 Protection mechanisms for file transfer 51
3.5.4 Protection mechanisms to ensure file consistency 52

3.6 Monitoring and controlling FT operation 54
3.6.1 FT logging 55
3.6.2 Console messages for automatic monitoring 56
3.6.3 SNMP management for *open*FT 58
3.6.3.1 Starting and stopping *open*FT 58
3.6.3.2 System parameters 59
3.6.3.3 Public key for encryption 59
3.6.3.4 Statistics 60
3.6.3.5 Diagnostic control 60
3.6.3.6 Partner Information 61
3.6.3.7 Traps 62
3.6.3.8 Trap groups and trap controls 62
3.6.3.9 Trap information 63

3.7 Administrating and controlling FTAC functions 64
3.7.1 Creating a default admission set 64
3.7.2 Administrating admission sets 64

3.7.3 Administrating admission profiles . . . 65
3.7.4 Transfer FTAC environment - the environment functions . . . 67
3.7.5 The FTAC logging function . . . 69

3.8 Using *open*FT in a HIPLEX cluster . . . 71

3.9 Diagnostics . . . 74
3.9.1 Controlling the trace function . . . 74
3.9.2 Evaluating traces . . . 76
3.9.3 Creating diagnostic records . . . 77

4 Administration commands . . . 79

4.1 Functional command overview . . . 80
4.1.1 FT commands . . . 80
4.1.2 FTAC commands . . . 82

4.2 Entering FT commands . . . 83

4.3 SDF syntax representation . . . 85

4.4 Command return codes . . . 102

4.5 OPS variables . . . 103

4.6 Output in CSV format . . . 104

4.7 ADD-FT-PARTNER (FTADDPTN)
Add remote system to the partner list . . . 105
4.7.1 Notes on entering partner systems . . . 110
4.7.2 Sample *open*FT partner entries . . . 110
4.7.3 Sample FTAM partner entries . . . 113
4.7.4 Partner systems linked via *open*FTIF . . . 115

4.8 CANCEL-FILE-TRANSFER or CNFT/NCANCEL/FTCANREQ
Cancel FT requests . . . 117

4.9 CREATE-FT-INSTANCE
Create a new *open*FT instance or activate an unloaded *open*FT instance . . . 123

4.10 CREATE-FT-KEY-SET (FTCREKEY)
Create a key pair set . . . 125

4.11 CREATE-FT-PROFILE
Create admission profile . . . 127

4.12 DELETE-FT-INSTANCE
Delete the administration entry of an *open*FT instance . . . 149

4.13 DELETE-FT-KEY-SET (FTDELKEY)
Delete a key pair set 150
4.14 DELETE FT-LOGGING-RECORDS (FTDELLOG)
Delete logging records 152
4.15 DELETE-FT-PROFILE
Delete admission profile 156
4.16 EXPORT-FTAC-ENVIRONMENT
Export FTAC admission profiles and sets 159
4.17 IMPORT-FTAC-ENVIRONMENT
Import FTAC admission profiles and sets 161
4.18 MODIFY-FILE-TRANSFER (FTMODREQ)
Modify request queue 164
4.19 MODIFY-FT-ADMISSION-SET
Modify admission set 169
4.20 MODIFY-FT-INSTANCE
Modify an *open*FT instance 176
4.21 MODIFY-FT-OPTIONS (FTMODOPT)
Modify operating parameters 178
4.22 MODIFY-FT-PARTNER (FTMODPTN)
Modify partner properties in the partner list 194
4.23 MODIFY-FT-PROFILE
Modify admission profile 199
4.24 REMOVE-FT-PARTNER (FTREMPTN)
Remove remote system from partner list 223
4.25 SHOW-FILE-TRANSFER or SHFT/NSTATUS/FTSHWREQ
Query status of FT request 224
4.26 SHOW-FTAC-ENVIRONMENT
Display saved admission profiles and sets 240
4.27 SHOW-FT-ADMISSION-SET
Display admission sets 244
4.28 SHOW-FT-LOGGING-RECORDS (FTSHWLOG)
Display logging records 249
4.28.1 Description of the output fields 256
4.29 SHOW-FT-OPTIONS (FTSHWOPT)
Display operating parameters 263

4.30 SHOW-FT-PARTNERS (FTSHWPTN)
Display partner systems 271

4.31 SHOW-FT-PROFILE
Display admission profile 279

4.32 SHOW-FT-RANGE
Display partner systems 285

4.33 START-FT (FTSTART)
Activate *open*FT 288

4.34 START-OPENFTPART
List partner systems as command procedure 290

4.35 STOP-FT (FTSTOP)
Deactivate *open*FT 291

4.36 UPDATE-FT-PUBLIC-KEYS (FTUPDKEY)
Update public keys 292

5 Appendix 293

5.1 Structure of CSV outputs 293
5.1.1 Output format 293
5.1.2 SHOW-FILE-TRANSFER 294
5.1.3 SHOW-FT-ADMISSION-SET 297
5.1.4 SHOW-FT-LOGGING-RECORDS 298
5.1.5 SHOW-FT-OPTIONS 300
5.1.6 SHOW-FT-PARTNERS 302
5.1.7 SHOW-FT-PROFILE 303
5.1.8 SHOW-FT-RANGE 304
5.1.9 SHOW-FTAC-ENVIRONMENT 304

5.2 Accounting records 305

5.3 Recovering from hung FT and FTAC subsystems 310

Glossary 311

Abbreviations 335

Related publications 341

Index 345

1 Introduction

The *open*FT product range transfers and manages files

– automatically,

– securely, and

– cost-effectively.

The reliable and user-friendly transfer of files is an important function in a high-performance computer network. Most corporate topologies today consist of networked PC workstations, which are additionally linked to a mainframe or Unix server. This allows much of the processing power to be provided directly at the workstation, while file transfer moves the data to the mainframe for further processing there as required. In such landscapes, the locations of the individual systems may be quite far apart. Fujitsu Siemens Computers offers an extensive range of file transfer products - the *open*FT product range - for the following system platforms:

- BS2000/OSD®,
- Solaris™(SPARC®/Intel™), LINUX®, AIX®, HP-UX®, OSF1(Tru64)
- Microsoft® Windows XP™, Windows Server 2003™, Windows Vista™
- OS/390 resp. z/OS (IBM®).

1.1 Brief description of the product *open*FT

*open*FT for BS2000/OSD is the file transfer product for computers using the operating system BS2000/OSD.

All *open*FT products from Fujitsu Siemens Computers intercommunicate via *open*FT protocols (originally: FTNEA protocols), which were standardized by Siemens. Since a number of FT products from other software suppliers also support these protocols, many interconnection options are available.

*open*FT also supports the FTAM file transfer protocol (File Transfer Access and Management) standardized by ISO (International Organization for Standardization). This makes it possible to interconnect to systems of other manufacturers whose FT products also support the same standard.

With the integrated FTAC function, *open*FT offers extended admission and access protection (FTAC stands for **F**ile **T**ransfer **A**ccess **C**ontrol).

*open*FT allows the use of TCP/IP, ISO TP0/2, ISO TP4, SNA and NEA as transport protocols.

*open*FT also possesses the add-on product *open*FT-FTP which supports ftp functionality.

1.2 Target group and objectives of this manual

This manual is intended for FT administrators and FTAC administrators.

To understand this manual, it is necessary to have a knowledge of the BS2000/OSD operating system.

1.3 Concept of *open*FT for BS2000/OSD manuals

The complete description of *open*FT and its optional components *open*FT-FTAM for BS2000, *open*FT-FTP for BS2000 and *open*FT-AC for BS2000 is contained in four manuals. In addition to this System Administrator Guide, there is also a User Guide, a Programmer Reference Guide and a Message Manual. The description is divided between the four manuals as follows:

- *open*FT for BS2000 Installation and Administration

  The System Administrator Guide is intended for FT and FTAC administrators. It describes:
  - the installation of *open*FT and its optional components
  - operation, control and monitoring of the FT system and the FTAC environment
  - the administration commands for FT and FTAC administrators
  - account records

- *open*FT for BS2000 Enterprise File Transfer in the Open World

  The User Guide contains the following information:
  - an overview of the basic functions of the *open*FT product family
  - a detailed description of the conventions for the file transfer to computers with different operating systems
  - information on the implementation of FTAM
  - description of the user commands
  - messages from *open*FT and *open*FT-AC

- *open*FT for BS2000 Programming Interfaces

  The Programmer Reference Guide describes the *open*FT and *open*FT-AC program interfaces.

- *open*FT for BS2000 Messages

  The manual contains the BS2000 messages of the FT system.

You will also find current information and runtime application scenarios on the Internet under *http://www.fujitsu-siemens.com/openft.*

1.4 Organization of the System Administrator Guide

This System Administrator Guide describes the command interface and tools available to FT and FTAC administrators. It is divided into five chapters.

The first chapter describes the layout of this manual and the changes introduced in *open*FT V10.0 for BS2000 as compared to the previous version V9.0.

The second chapter describes the installation of *open*FT for BS2000 and the prerequisites for using this product.

The third chapter describes the operation, control and monitoring of *open*FT and *open*FT-AC. It discusses the logging function, the SNMP connection, how to optimize the operating parameters, and what to do in the event of errors.

The fourth chapter describes the administration commands that are used by the FT/FT-AC administrator as tools in discharging his or her administrative duties.

The appendix contains a description of the command output in CSV format, an explanation of the FT accounting records. and the *open*FT console messages.

1.5 Changes since the last version of the manual

The following changes have been introduced in the *open*FT V10.0 for BS2000 System Administrator Guide since the earlier version *open*FT V9.0 for BS2000:

Extended support for character sets and character codes (CCS)

*open*FT supports Unicode, i.e., it is possible to exchange Unicode files with partners as of *open*FT V10. The Unicode CCSs are made available via XHCS and can be assigned for the local and for the remote files when text files are transferred.

The CCS is output with the SHOW-FILE-TRANSFER command.

Partner list and partner addressing

- The partner list replaces the network description file which was used in the past. The partner list and request queue are automatically set up on installation. The CREATE-FT-REQUEST-FILE and DELETE-FT-REQUEST-FILE commands therefore no longer exist. Entries are made in the partner list in the same way as for the earlier network description using the ADD-FT-PARTNER and MODIFY-FT-PARTNER commands.
- The partner address specification has been adapted to comply with the conventions for internet host names. When the address is specified, the employed protocol is specified at the same time (*open*FT, FTAM or ftp, see below). The syntax of ADD-FT-PARTNER and MODIFY-FT-PARTNER has been changed.
- In a request (file transfer, file management, FTAC profiles), partners can either be addressed directly via their address or via their names in the partner list. Consequently, the previous parameter PARTNER-NAME has been renamed to PARTNER. However, PARTNER-NAME can still be used as an alias name.

  The anonymous partner entry ANYBODY is only supported if the dynamic partner function is disabled. If dynamic partners are enabled then any partner which can be accessed via the employed transport system can also be addressed via *open*FT. It does not therefore have to be entered in the partner list.

Support for the ftp protocol

*open*FT supports file transfer and file management with ftp servers. If *open*FT is also used as the ftp server in the remote system then FTAC functions, pre-processing and post-processing can also be used.

In the case of ftp, partners can also be addressed via their IPv6 addresses.

Command interface for POSIX

The command interface originally used in *open*FT for UNIX systems and *open*FT for Windows systems is now also available for POSIX. To this end, the library SINLIB.OPENFT.100 must be installed in POSIX.

Other changes

- Message output has been modified (new message numbers, new texts).
- Authentication can now be forced on a partner-specific basis (MODIFY-FT-PARTNER).
- The operating parameters REQUEST-WAIT-LEVEL, PACING and MAX-INBOUND-REQUEST can no longer be modified.
- Traces can be controlled as a function of partner type (*open*FT, FTAM, ftp) and request type (local/remote and synchronous/asynchronous). The syntax of MODIFY-FT-OPTIONS and MODIFY-FT-PARTNER has therefore been modified.
- The TASK-LIMIT parameter of the MODIFY-FT-OPTIONS command has been renamed to PROCESS-LIMIT and no longer supports the specification 0.
- The default value for INITIATOR in CREATE-FT-PROFILE has been changed from *REMOTE to (*LOCAL,*REMOTE).

1.6 README file

Information on any functional changes and additions to the current product version described in this manual can be found in the product-specific README file. You will find the README file on your BS2000 computer under the file name

```
SYSRME.OPENFT.100.E
SYSRME.OPENFT-FTAM.100.E
SYSRME.OPENFT-FTP.100.E
SYSRME.OPENFT-AC.100.E
```

The user ID under which the README file is cataloged can be obtained from your system administration. You can view the README file using the SHOW-FILE command or an editor, and print it out on a standard printer using the following command:

```
/PRINT-DOCUMENT filename,LINE-SPACING=*BY-EBCDIC-CONTROL
```

2 Installation and startup

This chapter describes the actions and preconditions required to install and run *open*FT and any of the optional components *open*FT-FTAM, *open*FT-AC, *open*FT-FTP and *open*FT-CR in BS2000.

2.1 Installing *open*FT

*open*FT V10.0 requires the following software

- BS2000/OSD as of V5.0
- *open*Net server as of V3.0 (i.e. BCAM ≥ V17.0)
- SDF ≥ V4.0

The following versions are required when using the optional add-on components:

- *open*FT-FTAM V10.0
- *open*FT-AC V10.0
- *open*FT-CR V10.0
- *open*FT-FTP V10.0

If you want to make use of the POSIX functionality, you will also need the BS2000/OSD component POSIX.

*open*FT-FTAM V10.0 requires:

- *open*FT ≥ V10.0
- OSS ≥ V4.1

*open*FT-AC, *open*FT-FTP and *open*FT-CR V10.0 require:

- *open*FT ≥ V10.0

The graphical administration interface requires:

- Windows 2000, Windows Server 2003 or Windows XP
- Desk2000 ≥ V4.0
- *open*FT ≥ V7.0 for BS2000

Delivery of *open*FT is done via the software delivery and information system SOLIS2. Installation is done via IMON. The installation routine incorporates the required BS2000-specific tasks such as the MSGFILE update, subsystem catalog entries and the integration of the SDF syntax file.

Whenever a program which uses FT interfaces is compiled, the file SYSLIB.OPENFT.100 for COBOL and ASSEMBLER programs must be available. This file must be available as a shareable file in the system, but need not be located under the TSOS ID.

2.1.1 Initial installation of *open*FT for BS2000/OSD

*open*FT is a subsystem and is not generated when the BS2000 system is generated.

The FT administrator commands can be issued from the console. Administration from the terminal requires the FT-ADMINISTRATION privilege, assigned by default to TSOS. If SECOS is in use, this privilege can be assigned to other user IDs. See the SECOS manual for details.

In order to ensure the usability of the COBOL program interface, the file SYSRTC.FT (runtime module for the COBOL program interface) must be shareable under the SYSFJAM ID (`SHARE=YES, ACCESS=READ`). COBOL programs produced with the COBOL interface load the runtime module from this ID.

The product PAMINT is used to convert from keyed to nonkey files (and vice versa). This product belongs to the BS2000 basic configuration and must be available under TSOS. PAMINT should be installed with IMON so that *open*FT can automatically load the current version and use it for conversion. If this is not possible, then *open*FT uses the file $TSOS.SYSLNK. PAMINT or $TSOS.SYSREP.PAMINT at link time. In this case, these files must therefore contain copies of the current PAMINT SYSLNK or SYSREP file.

2.1.2 Version change and compatibility

*open*FT V10.0 is, with the following exceptions, compatible with *open*FT V9.0.

- The messages have been completely revised. As a result, most message numbers have changed, some messages have been removed and new messages have been added.
- The previous request queue and network description file no longer exist in their previous form since the files for request and partner administration are set up by *open*FT on installation. This has the following consequences:
  - The CREATE-FT-REQUEST-FILE and DELETE-FT-REQUEST-FILE commands no longer exist.
  - Instead of a network description file, *open*FT now sets up a partner list.
  - The entries in the partner list are optional, i.e. can optionally also be accessed directly by means of their addresses without having to be entered in the partner list.
- The operating parameters REQUEST-WAIT-LEVEL, PACING and MAX-INBOUND-REQUEST can no longer be modified.

For further details, refer to the release notice.

2.1.3 Installation of the command interface for POSIX

As of *open*FT V10.0, the command interface which is provided in UNIX and Windows systems is also supported in POSIX. This means that you can use the *open*FT functions in BS2000 from within a POSIX system. WIth a very few exceptions, the commands have the same function scope as they do in UNIX or Windows systems.

For details see the user guide and the administrator guide of *open*FT for UNIX Systems.

Installation

The library SINLIB.OPENFT.100 is required for installation. This contains the installation script and all the components needed for POSIX installation. The POSIX subsystem must be active for installation.

You install the command interface as follows:

– Call the START-POSIX-INSTALLATION command.

– Select “Install packages in POSIX” from the menu and enter the data in the “BS2000 POSIX package installation” screen which now opens. For details, see the POSIX manual “Basics for Users and System Administrators”.

2.2 Startup

The FT administrator's tasks have been simplified in *open*FT V10.0 since the request queue and partner list files are created when *open*FT is installed.

2.2.1 Preparing the FT system

For the first installation, an ID with the name SYSFJAM must be created for *open*FT on the home pubset of the processor. If you are running multiple *open*FT instances on your system, you must set up the configuration user IDs of the instances so that they are the same as SYSFJAM. The IDs should be set up in a manner that prevents a SET-LOGON-PARAMETERS command being entered. The default catid must be on the home PVS. The number of PAM pages required by this ID depends on:

– the size of the request files and partner lists used and the required functionality, i.e., whether FTAC functionality is to be used. The *open*FT request queue SYSRQF has a default size of 12864 PAM pages while the option file SYSOPF requires 12 PAM pages and the partner list 4992 PAM pages. The FTAC file SYSFSA occupies at least 501 PAM pages.

– the size of the log file (at least 501 PAM pages) which, in turn, depends on the number of transfer requests handled and on which sets are engaged (i.e. only FT sets, only FTAC sets, or both).

– the size of the SYSFSI, SYSKPL, and SYSKEY files (by default, a total of 69 PAM pages), and on the number of key pairs (SYSPKF files) created.

– the type and number of trace functions activated.

It is therefore advisable to allow PAM page overruns for the ID SYSFJAM.

In order to be able to access public keys, the FT administrator needs access to the SYSPKF files and the SYSKEY library on SYSFJAM or on the configuration user ID. If he/she does not have privileges granting him/her access to operating system resources, the FTAC admissions profiles should be set up to grant him/her access.

*open*FT requires a subsystem catalog entry containing a subsystem declaration with the load time set to "AT-CREATION-REQUEST". The FT subsystem must be explicitly loaded in a startup procedure (e.g. CMDFILE).

When an FT instance is stopped, (particularly by using /STOP-SUBSYSTEM FT) all the file locks held by *open*FT (see page 52) are cleared and, on loading an instance (e.g. by using /START-SUBSYSTEM FT), the locks are reset for files affected by existing requests. The FT or system administrator must therefore observe the following:

– On starting the FT subsystem, all pubsets which contain files that may need to be transferred by *open*FT must be available, since *open*FT will otherwise be unable to set the file lock, thus causing the related transmission to be aborted. On the other hand, the loading must also occur early enough to ensure that the files to be transmitted are protected in time.

– Unloading an FT instance should be done as late as possible, but before the export of the pubsets on which the files to be transmitted are located.

If desired, the FT logging function must be enabled with MODIFY-FT-OPTIONS before starting *open*FT (with /START-FT)

The job class JBCLLST should be generated with a small maximum processing time and, if possible, a high selection priority for printing result lists. This job class should be accessible to all FT users. The high priority (JOBPRIORITY operand in job class setup) ensures that jobs of this type are quickly started. A low maximum processing time (CPU-TIME operand in job class setup) prevents these jobs blocking the processor for a prolonged time.

For follow-up processing initiated by the *open*FT, you should generate the job class JBCLJOB with low maximum processing time and, if necessary, a high selection priority. If you do not do this the default job class will be used for follow-up processing. You should start extended, CPU-intensive follow-up jobs as enter jobs using the job classes which are available as standard in the BS2000-System.

As an alternative to port 102, *open*FT for BS2000 can also be reached by port 1100 (*open*FT protocol) or port 4800 (FTAM protocol). To do this, *open*FT itself creates a BCMAP entry on /START-FT. The following command is set for initializing mapping:

/BCMAP FUNCT=INIT,MAXMAP=500

If initialization is to be done using other values, it must take place before the first /START-FT command.

2.2.2 Entering partners in the partner list

In *open*FT V10.0, the network description has been replaced by a partner list. The partner list is set up by *open*FT on installation. Following a new installation, it is empty.

Although the entry of partners in the partner list is optional as of *open*FT V10.0, this offers significant advantages. These include simplified addressing for users, the central administration of partner addresses and enhanced security since you can assign individual properties such as security level or partner check level to partner systems, for example if authentication is required. Consequently, you should enter partners with special characteristics in the partner list immediately after installation. The following options are available:

– If you are upgrading from an older *open*FT version, start the command procedure which you created with START-OPENFTPART or SHOW-FT-PARTNERS in the older *open*FT version. The entries from the previous network description are taken over into the partner list.

– ADD-FT-PARTNER command
  This enters a new partner in the partner list.

In the operating parameters, you can specify that only partners from the partner list may be addressed (corresponds to the state up to *open*FT V9.0).

For further details on administering partners during operation, see section “Administering partners” on page 38.

2.2.3 Starting and stopping *open*FT

Starting *open*FT

*open*FT is started with the START-FT command. Care must also be taken to ensure that all pubsets are available, as otherwise any locally submitted request that requires an unavailable pubset will be terminated with an error message. If this happens, the user cannot be informed of the circumstances by an event list or a job variable.

The FTAM and FTP components (if installed) are activated on START-FT.

If multiple instances are used on one computer, each instance must be started individually. Individual instances can be set up so that they are automatically started on executing the command START-SUBSYSTEM, see the section “Using openFT in a HIPLEX cluster” on page 71).

If the HOST-NAME option is not set at start time then the real BCAM host is used. If multiple instances have to be started in a system then the host name must be set using the /MODIFY-FT-OPTIONS command at all but one of them.

Stopping *open*FT

Using STOP-FT terminates *open*FT in the current instance. When file transfer is terminated, non-restartable requests are aborted. Local requests continue to be accepted even after STOP-FT. The requests are stored in the request queue until *open*FT is restarted. When START-FT is entered again, the requests are processed in sequence.

2.3 Installing *open*FT-AC

The installation of *open*FT V10.0 is required for the installation of *open*FT-AC V10.0.

2.3.1 Initial installation

Delivery of *open*FT-AC takes place using the software delivery and information system SOLIS2. Installation takes place via IMON. If required, the installation contains BS2000-specific jobs such as MSGFILE update, subsystem catalog entries, and importing the SDF syntax file.

For the security of the SYSFSA file on the configuration user ID of the current *open*FT instance, it is recommended that you activate the class 2 ENCRYPTION option for password encryption. This file contains the settings for admissions sets and admissions profiles.

2.3.2 Version change

If an older version of *open*FT-AC is installed on your computer, it is recommended that you delete all product files of the old version. Profiles and admission sets from this version can be transferred unchanged. If you want to have the option of subsequently reverting to the earlier version then you should back up the file $SYSFJAM.SYSFSA.

2.4 Installing the graphical interfaces

System requirements:

- Windows 2000, Windows Server 2003 or Windows XP
- Desk2000 ≥ V4.0
- *open*FT ≥ V7.0
- If the emulation MT9750 is to be used instead of DoorsEm by Desk2000, you will need MT9750 version 4.0 or higher.

In order to install "*open*FT and *open*FT-AC for BS2000", DESK2000 and an emulation must be installed first, and a session with the BS2000 host must be established. (The default Desk2000 installation also automatically installs an emulation.)

When you are installing DESK2000, note that at least the SDF-DOORS component must be installed. As far as possible, we recommend that you perform the standard installation.

After installing DESK2000 and the emulation, you will need to set up a session with the BS2000 host and save this session as a .MTS file (for MT9750) or a .DRK file (for the DOORS emulation). If you are using the MT9750 emulation, the terminal must be configured as a terminal of type 9763.

You can then install "*open*FT and *open*FT-AC for BS2000" from the medium on which it was delivered.

Installation occurs via a convenient interactive installation program. Enter the data medium and start the Setup program for the installation as usual in Windows.

After calling the setup program, you can choose to install *open*FT and/or *open*FT-AC (in German or English). You can click the "Readme" button to obtain additional information. You should also read the license conditions which can be displayed by clicking the "License" button. As installation continues, you can decide the folder and program group to which the program is to belong.

2.4.1 Connection setup

Communication with BS2000 occurs via SDF-DOORS.

If you are setting up a connection for the first time, use the command “New” from the “Session” menu. This brings up the SDF-DOORS window “Create a New Connection”. Enter the .EXE file of the emulation to be used (MT9750.EXE or DRWS_EM.EXE) in the “Command Line” field and enter the associated session file (*.MTS or *.DRK). in the “Connection Parameters” field. When you then click “OK”, the *open*FT dialog box “Logon” window will appear. You can then enter the LOGON parameters for the connection in this window.

When SECOS is used, users are able to protect their user IDs with an additional ID and password in BS2000/OSD (SET- or MODIFY-LOGON-PROTECTION). When issuing a LOGON command, the user must then also specify this ID and the associated password. For this reason, the “Set Personal Attributes”, “User-Identification” and “Password” boxes have been added to the “Logon” dialog box.

The following entries can be made in the “Logon” dialog box:

User-Identification: Enter the user ID under which you want to administer *open*FT here.

Account: Account number of the user ID.

Password: Here you enter the BS2000 password of the user ID under which you want to administer *open*FT. An asterisk (*) is displayed for each character entered. The password must be enclosed in single quotes. If the user ID has no password, the default value displayed in this field, i.e. *NONE (*****), must not be deleted. If the specified user ID has a password and you do not enter it, i.e. you leave the default password *NONE in the field, the emulation waits for the hidden entry of the password.

Job Name: Specification of the job name.

Save Logon Parameters: If you check this field (a cross appears in the check box), the LOGON parameters can be saved in an SDS file, which can be created with the “Save As” command from the “Session” menu. When you subsequently open a session using that SDS file, the LOGON saved in it is executed automatically.

Set Personal Attributes: If you check this field (a cross appears in the check box), you can protect this user ID with a further ID and associated password.

User-Identification: Specify the user ID for your personal identification.

Password: Specify the password of the user ID for your personal identification. Each character entered is displayed as an asterisk (*). The password must be enclosed in single quotes.

This establishes the connection to BS2000. After setting up the connection, you can save this session in an SDS file by using the “Save As” command from the “Session” menu. This SDS file can then be specified explicitly with the “Open” command for subsequent calls to *open*FT or entered for the “Automatic Connect” command in the “Options” menu. In the latter case, the next connection will be automatically established using the entered SDS file.

If you want to speed up and simplify the connection setup even further, you can use SDF-DOORS to embed the BS2000 syntax file in the SDS file. To do this, first call SDF-DOORS and establish a connection to BS2000 using the SDS file. Then call the “Edit Session” command from the “Options” menu. You can now enter an offline syntax file (which must be available locally on the PC) in the “Syntax File” field and save it in the parameter file. Enter the Logon command in the “Startup Command” field. After closing the connection in SDF-DOORS, save the session (in an SDS file). See also the SDF-DOORS documentation for more details. You can then start this function with the “Syntax File” command in the *open*FT “Options” menu.

2.4.2 Configuration

*open*FT is configured by default in the registry, but you can also create a separate configuration file for each application. These files are stored in the Windows directory. The pathname is freely selectable and can be specified in the Start command of “*open*FT for BS2000” and “*open*FT-AC for BS2000”. If the specified file does not exist, it will be created under the specified name, and if no file is specified, the default name is used.

2.5 Configuring *open*FT-AC

Authorization of the FTAC administrator

It is recommended that the position of administrator for *open*FT-AC be given to a user in the system who is responsible for data protection in a BS2000 system, since he will know what protection measures are required where.

The FTAC administrator function is assigned by means of the SECOS privilege FTAC-ADMINISTRATION. It may also be assigned to several user IDs at once.
For BS2000 installations without SECOS, the administration attribute has a fixed assignment to the user ID TSOS.

FTAC administrators who possess both the FTAC administration and TSOS privilege have the following additional rights:

– If they import profiles (for any user ID), they can select whether the profiles will be immediately available and unrestricted, or whether they will be locked.
– If they create profiles for external IDs then these are also immediately available. This means that they can create valid transfer admissions even if they do not know the LOGON password of the target ID. This method can be used to set up profiles that remain valid after the LOGON password is modified.
– They can therefore also modify the transfer admissions of existing profiles with external IDs without knowing the profile owner's password.

Adapting the default admission set

After the installation of *open*FT-AC, all values of the default admission set are set at 0!

This means that it is not yet possible to execute a file transfer with the local system. This is because as long as no other admission sets are made with MODIFY-FT-ADMISSION-SET, the default admission set is valid for all user IDs. The maximum security level 0 for the basic functions means that these basic functions may not be used. The FTAC administrator must therefore use the command MODIFY-FT-ADMISSION-SET to raise the values of the default admission set.

Default security levels for partners

The FT administrator can use the MODIFY-FT-OPTIONS command (SECURITY-LEVEL operand) to define default security levels for all the partner systems entered in the partner list. The administrator can either enter a fixed value or specify *BY-PARTNER-ATTRIBUTES to indicate that the security level is set automatically: partners which are authenticated by *open*FT are assigned security level 10. Partners which are known in BCAM (i.e. they are addressed via their BCAM name) are assigned security level 90. All other partners are assigned security level 100.

This automatic assignment can also be activated on a partner-specific basis using the operands of the same name:
ADD-FT-PARTNER and MODIFY-FT-PARTNER...,SEC-LEV=*BY-PART-ATTR

This automatic assignment always applies to partners that are not in the partner list.

Examples

1. All partner systems should be accessible for file transfer for all FTAC users. This is achieved by setting all the values of the default admission set to100. The following command is used:

   ```
   /MOD-FT-AD␣*STD,MAX-LEV=100
   ```

   More information on the command MODIFY-FT-ADMISSION-SET can be found starting on page 169.

2. A differentiated setting of the default admission set might look as follows:

   ```
   /MODIFY-FT-ADMISSION-SET USER-IDENTIFICATION=*STD          -
                            MAX-LEVELS=(OUTBOUND-SEND=50,     -
                                        OUTBOUND-RECEIVE=50,  -
                                        INBOUND-SEND=20,      -
                                        INBOUND-RECEIVE=20,   -
                                        INBOUND-PROCESSING=10, -
                                        INBOUND-MANAGEMENT=0)
   ```

   The different security levels are assigned selectively. For example, the function “inbound management” can be fully blocked by setting the security level to 0.

   WARNING!

   Note that *open*FT-AC is only effective for connected products such as *open*FT or FTP. If other file transfer products without an *open*FT-AC connection are also being used, a more comprehensive and coordinated security concept would be advisable.

3 Operation

This chapter contains information on the subject of administration, security and control and monitoring functions.

FT and FTAC administration

An FT user can monitor and administer only his or her own FT requests, whereas the FT administrator has access to all FT activities occurring in his or her system.

The FTAC administration is independent of the FT administration. The FTAC administrator is the security manager of FT activities in your computer. He has “ultimate authority” over all admission sets and profiles.

If you also have SECOS in use, you will require the privilege FT-ADMINISTRATION for FT administration and FTAC-ADMINISTRATION for FTAC administration. In other cases, the system administrator ID TSOS must be used.

FTAC administrators who possess both the FTAC administration and TSOS privilege have the following additional rights (see section “Configuring openFT-AC” on page 26).

*open*FT V10 for BS2000 can be administered via the graphical user interface for *open*FT V10 for Windows or *open*FT V10 for UNIX systems. It is possible to process the request queue, admission profiles, admission sets, logs and the partner list.

Administration via the graphical user interface

Using the product DESK2000 as a basis, *open*FT and *open*FT-AC for BS2000 can now be conveniently administered via a graphical interface from a Windows workstation.

This graphical interface enables you to view requests and manage them (i.e. prioritize, delete) as required; to create templates for transfer requests and manage them. The requests to be displayed can be selected using different criteria. File transfer with multiple file selection is also possible. You can also display logging records (all users) and delete these (administrators only).

If you log on under a user ID with administrator privileges, you can also execute administration functions for *open*FT for BS2000 via the interface (i.e. administer partners, enable and disable tracing, evaluate traces, start and stop *open*FT, set FT options, etc.).

Depending on your logon authorization (normal user or FTAC administrator), you can also administer admission sets and profiles, including those which are saved externally.

The appropriate documentation for this purpose, especially with respect to the embedding of the interface in Windows and the interaction with DESK2000, can be found in the respective online help. The significance of the individual dialog box elements corresponds to the commands described in this manual.

3.1 Optimizing the operating parameters

The proposals listed below suggest a number of ways in which the FT administrator can optimize FT operation by modifying the operating parameters. It is always advisable to alter only one operating parameter at a time, so that the precise effects of the change can be observed.

3.1.1 Interdependencies for optimized parameterization

The optimum settings for operating parameters depend on several different constraints:

- load levels of the local and remote systems,
- load level in the network,
- line transfer rates in the network,
- network structure (connection paths reserved for FT or shared paths for FT and dialog),
- incorporation of gateway computers (e.g. TRANSIT),
- type, performance or generation of the transport system used,
- average size of files to be transferred,
- number of files to be transferred (e.g. per day).

WARNING!

In the case of TCP/IP networks, the RFC1006 protocol variant is significantly superior to the older CVP protocol (LANINET) in both throughput and resource consumption.

In some instances, these boundary conditions are themselves subject to dynamic change (load levels for example), so it is not possible to calculate in advance the optimized values for a particular installation.

3.1.2 Achieving optimized operation

Experience has shown that the most suitable parameter settings can only be achieved in stages.

Initially the *open*FT default values should be left unchanged. In most cases it will be possible to run file transfers satisfactorily using these parameter values.

If not, however, as a second step an improvement can be sought by changing **one** of the parameter values. It is normally not advisable to change more than one parameter at a time as otherwise there is no way of ascertaining the precise effect of each change.

If satisfactory operation of the FT system has still not been achieved, the FT administrator can repeat the second step, changing a different parameter.

The FT administrator can control the operation of the FT system using the parameters PROCESS-LIMIT, CONNECTION-LIMIT, TRANSPORT-UNIT-SIZE and MAX-REQUEST-LIFETIME, see the following table:

Problem	**Suggested solution**
Poor dialog response times	1. Reduce TRANSPORT-UNIT-SIZE 2. Reduce CONNECTION-LIMIT
Computer overloaded, network load not yet optimized	1. Set PROCESS-LIMIT to 2 2. Increase TRANSPORT-UNIT-SIZE 3. Reduce CONNECTION-LIMIT
Computer and network overloaded	1. Set PROCESS-LIMIT to 2 2. Reduce CONNECTION-LIMIT
Throughput inadequate	1. Increase TRANSPORT-UNIT-SIZE
Prolonged requests block other requests	1. Increase CONNECTION-LIMIT
Requests to a particular partner system use up all resources	1. Increase CONNECTION-LIMIT
Requests from partner systems (inbound requests) use up all resources	1. Increase CONNECTION-LIMIT
Requests are present in the request file for a very long period without being processed.	1. Set MAX-REQUEST-LIFETIME

The command used for this purpose is MODIFY-FT-OPTIONS. These parameters are discussed in the sections below. In addition, the effect of changing the parameters is also described.

3.1.3 Changing the PROCESS-LIMIT operating parameter

The PROCESS-LIMIT parameter defines the maximum number of tasks that may be used for processing file transfer requests. The number of file transfer requests per task handled simultaneously can be expressed as follows:

$$\frac{\text{CONNECTION-LIMIT}}{\text{PROCESS-LIMIT}}$$

CONNECTION-LIMIT is the maximum number of parallel transport connections that can be used to execute requests.

If the PROCESS-LIMIT value remains fixed and the value of CONNECTION-LIMIT is increased, then proportionately more transport connections are available for each task and therefore more requests can be processed per task. The reduction of the PROCESS-LIMIT value where CONNECTION-LIMIT remains constant achieves the same effect. If the value of the quotient is reduced (by reducing CONNECTION-LIMIT or increasing PROCESS-LIMIT), a smaller proportion of transport links is available per task. Consequently, fewer requests can be processed per task.

If the number of requests awaiting processing exceeds the value of the quotient but the number of tasks assigned has not reached the PROCESS-LIMIT value, then another task is initiated.

The setting PROCESS-LIMIT=*NONE corresponds to the setting PROCESS-LIMIT= CONNECTION-LIMIT. A separate task is generated for each connection.

Higher PROCESS-LIMIT:

– fewer wait times for input/output

– better use of potentially underutilized computer resources

Lower PROCESS-LIMIT:

– reduced load on the local system

3.1.4 Changing the CONNECTION-LIMIT operating parameter

The CONNECTION-LIMIT parameter defines the maximum number of transport connections to be used in the execution of file transfer requests. Since the processing of a request always requires a new transport connection to be set up, CONNECTION-LIMIT also defines the maximum number of requests the system can process in parallel.

One third of the connections are reserved for requests from remote systems, thus preventing locally symbiotic requests blocking the system for remote requests. The second third is reserved for requests from the local system, and local and remote system requests compete for the last third.

In order to obtain the same level of throughput with your *open*FT partners, it may therefore be necessary to increase the CONNECTION-LIMIT value.

Higher CONNECTION-LIMIT:

– increased data throughput

– better use of potentially underutilized processor capacity.

Lower CONNECTION-LIMIT:

– reduced load on the local system and network, and hence less or even no impact upon interactive operation.

3.1.5 Changing the TRANSPORT-UNIT-SIZE operating parameter

The TRANSPORT-UNIT-SIZE parameter defines the maximum length of the message transmitted to the transport system by *open*FT. TRANSPORT-UNIT-SIZE has no effect for links to FTAM partners. Message flow control ensures that only a specific number of messages are being transmitted across the network at any one time. The TRANSPORT-UNIT-SIZE parameter enables the administrator to control the amount of FT data present in the network at a particular time. The value specified for TRANSPORT-UNIT-SIZE can be changed by the remote system or by the transport system (maximum message length). A maximum value of 65535 is recommended for TRANSPORT-UNIT-SIZE. This value is the default value after installation.

Higher TRANSPORT-UNIT-SIZE:

– increased data throughput

– reduced load on the local system since fewer calls to the transport system are necessary.

Lower TRANSPORT-UNIT-SIZE:

– reduced load on the network

– the time required to transmit an FT message across a communication link is reduced, which in turn decreases the wait time for messages from other users. For slow communication links, response times can, for example, be improved in interactive mode.

3.1.6 Setting the MAX-REQUEST-LIFETIME operating parameter

The MAX-REQUEST-LIFETIME parameter is used to set a global limitation for the lifetime of *open*FT requests. Whereas it was previously necessary to use CANCEL-FILE-TRANSFER to remove requests without a lifetime limitation from the request file, you can now specify a maximum lifetime (calculated from the start time). The maximum lifetime applies to both inbound and outbound requests and is specified in days.

3.2 Administering code tables

The concept of so-called “Coded Character Sets” (CCS) is supported for *open*FT partners as of V10. A CCS defines a character set and the coding of these characters in the file. A CCS is assigned a name of up to 8 characters in length via which the CCS can be addressed.

When transferring text files, users can specify a separate CCS for file encoding in the local and remote systems (as of *open*FT V10).

Frequently used Coded Character Sets are:

ISO88591	Character set in accordance with the definition contained in ISO standard 8859-1, ASCII-oriented coding in accordance with ISO standard 8859-1.
EDF041	Character set in accordance with the definition contained in ISO standard 8859-1, EBCDIC-oriented coding in accordance with FSC definition DF04-1.
UTF8	The character set is Unicode, the UTF-8 multi-byte coding defined in the Unicode standard is used.
UTF16	The character set is Unicode, the UTF16 16-bit coding defined in the Unicode standard is used.
CP1252	The character set is a superset, defined by Microsoft, of the character set defined in ISO standard 8859-1. The ASCII-oriented coding is identical to the ISO8859-1 for the characters which are shared with ISO8859-1. The other characters defined by Microsoft (including the Euro symbol) are present in the code range 0x80-0x9F which is not used by ISO8859-1.

Making a CCS available

In BS2000/OSD , the CCSs are defined and made available via XHCS. The default CCS for the system (HOSTCODE) is defined by the BS2000 system administrator. The administrator can also assign a default user character set different to HOSTCODE to a user ID. As FT administrator, you must consult with the BS2000 system administrator to ensure that the required code tables are available on the system.

On the other *open*FT platforms as of V10, the commonly used CCSs are supplied with *open*FT. The FT administrator defines the default character set via the operating parameters.

3.3 Administering requests

You can use the SHOW-FILE-TRANSFER command (see page 224ff) to view information on selected FT requests. Possible selection criteria include

– the user ID,

– the system which initiated the request,

– certain statuses of FT requests, and

– names of file or job variables affected by an FT request in the local system.

– the pubset on which the transfer files are located.

The MODIFY-FILE-TRANSFER command permits both administrator and user to modify the order and priority of outbound requests within the request queue.

The CANCEL-FILE-TRANSFER command enables you to remote FT requests from the request queue or to abort file transfer while in progress. The selection criteria at your disposal are much the same as those for the SHOW-FILE-TRANSFER command. In particular, the FT administrator can purposely delete requests that lock files on a certain pubset (for example, if pubsets are to be reconfigured).

The FT administrator can use the CANCEL-FILE-TRANSFER ... FORCE-CANCELLATION command to force the full, unconditional cancellation of a request and its removal from the request file, if necessary without any negotiation with the partner system.

MODIFY-FT-PARTNER allows you to activate or deactivate locally submitted requests for a particular remote system (see STATE, page 194).

3.4 Administering partners

*open*FT offers the FT administrator four commands for the administration of partner systems:

ADD-FT-PARTNER	Add new partner system entries to the partner list
MODIFY-FT-PARTNER	Modify partner system entries in the partner list
REMOVE-FT-PARTNER	Remove partner systems from the partner list
SHOW-FT-PARTNERS	View information on partner systems in the partner list and back up the partner list (page 43)
START-OPENFTPART	Back up the partner list (page 43)
MODIFIY-FT-OPTIONS	Enable/disable dynamic partners (page 43)

For links with FTAM partners FT-BS2000 assumes that the transport system permits parallel connections. The remote systems are identified via their presentation addresses. Either BS2000 or the FTAM partner can initiate file transfer.

3.4.1 Defining partner properties

You use the ADD-FT-PARTNER command to define the properties of partners:

- Partner address, see page 39
- FTAC security levels, see page 42
- Automatic deactivation, see page 42
- Partner-specific trace settings, see page 74
- Authentication setting and instance identification for the partner, see page 46
- Sender verification, see page 51

You can modify these settings whenever you want with MODIFY-FT-PARTNER.

3.4.1.1 Specifying partner addresses

The addressing of partner systems has been modified in *open*FT V10 in terms of both syntax and function scope:

– the structure of the partner address has been considerably simplified and adapted to comply with internet address conventions, see “Structure of the partner address”. You specify the partner address as in the past in the ADD-FT-PARTNER or MODIFY-FT-PARTNER command.

– a partner can be accessed directly via its address in FT requests even if it is not entered in the partner list. This is only possible if the “dynamic partner” function is enabled, see page 43. The partner entry ANYBODY is not used in this case.

– It is now also possible to address ftp partners.

Structure of the partner address

A partner address has the following structure:

[protocol://]host[:[port].[tsel].[ssel].[psel]]

host (= computer name or processor name, see page 39) is mandatory; all other specifications are optional. In many cases, the other specifications are covered by the default values, so that the host name suffices as the partner address, see “Examples” on page 41. Final ’.’ or ’:’ can be omitted.

The individual components of the address have the following meanings:

protocol://
Protocol stack via which the partner is addressed. Possible values for *protocol* (uppercase and lowercase are not distinguished):

openft *open*FT partner, i.e. communication takes place over the *open*FT protocol.

ftam FTAM partner, i.e. communication takes place over the FTAM protocol.

ftp FTP partner, i.e. communication takes place over the FTP protocol.

Default value: **openft**

host
Computer name via which the partner is addressed. Possible entries:

– internet host name (e.g. DNS name), length 1 to 80 characters, only for ftp partners
– BCAM processor name, length 1 to 8 characters
– SNA LU name, length 1 to 8 characters
– IPv4 address (only with FTP partners) with the prefix %ip, i.e. for example %ip139.22.33.44

– IPv6 address (only with FTP partners) with the prefix %ip6, i.e. for example `%ip6[FEDC:BA98:7654:3210:FEDC:BA98:7654:3210]` (ipv6) or `%ip6[FE80::20C:29ff:fe22:b670%5]` (ipv6 with scope ID)

  The square brackets [..] must be specified.

  The scope ID designates the local network card via which the remote partner can be accessed in the same LAN-Agment. It must be appended to the address with a % character. In Windows systems, this is a numerical value (e.g. 5). On other systems, it may also be a symbolic name (e.g. *eth0*). The scope ID can be identified using the *ipconfig* command.

port

When a connection is established over TCP/IP, you can specify the port name under which the file transfer application can be accessed in the partner system.
Permitted values: 1 to 65535;

In the case of an SNA-LU connection, (*host* = LU name) you must specify the value *sna* for the port number.

Default value: **1100** for *open*FT partners not connected via *open*FTIF

4800 for FTAM partners not connected via *open*FTIF

21 for FTP partners

1400 for connections via *open*FTIF with *open*FT or FTAM partners. In this case, the transport selector is FJMFTIF*n*.

tsel

Transport selector under which the file transfer application is available in the partner system. The transport selector is only relevant for *open*FT and FTAM partners.
You can specify the selector in printable or hexadecimal format (0xnnnn...). The specification will depend on the type of partner:

– *open*FT partner:
  Length 1 to 8 characters, a printable selector will be coded in EBCDIC in the protocol and may be padded with spaces internally to the length of eight characters.

  Default value: **$FJAM**

– FTAM partner:
  Length 1 to 16 characters; a printable selector will be coded as variable length ASCII in the protocol. Exception: The T-selectors $FTAM (default value) and FJMFTIF*n* are coded in EBCDIC and padded with spaces to the length of 8 characters.

  Default value: **$FTAM**

Note
With FTAM partners on Windows systems, you must generally specify the value **SNI-FTAM**.

ssel
Session selector under which the file transfer application is accessible in the partner system. You can specify the selector in printable or hexadecimal format (0xnnnn...).

Length 1 to 16 characters; a printable selector will be coded as variable length ASCII in the protocol.

Default value: empty

psel
Only relevant for FTAM partners.
Presentation selector under which the file transfer application is accessible in the partner system. You can specify the selector in printable or hexadecimal format (0xnnnn...).

Length 1 to 16 characters; a printable selector will be interpreted as variable length ASCII in the protocol.

Default value: empty

Examples

The partner computer with the host name FILESERV is to be addressed over different protocols/connection types:

Connection type/protocol	**Address specification**
*open*FT partner	FILESERV
*open*FTIF partner	FILESERV:.FJMFTIF0.TNSPART1
FTAM partner (BS2000 or UNIX system)	ftam://FILESERV
FTAM partner (Windows system)	ftam://FILESERV:.SNI-FTAM
Third-party FTAM partner	ftam://FILESERV:102.tsel.ssel.psel
FTP partner	ftp://FILESERV
SNA partner via *open*FT protocol (FILESERV is the LU name)	FILESERV:.sna

3.4.1.2 FTAC security levels for partner entries

If FTAC functionality is to be used, the FTAC administrator can use SECURITY-LEVEL in the partner entry to define the requisite security level for FTAC.

The security levels regulate the degree of protection with respect to the partner system. This protection can be best determined by the FTAC administrator. Therefore, he should advise the FT administrator on the assignment of the security levels to the partner systems. A high security level is used when a high degree of security is required, and a low level for a low degree of security. When FTAC is first installed, the security levels should be assigned in multiples of ten. This leaves the option open to incorporate new partner systems flexibly into the existing hierarchy.

If the degree of required security changes with respect to a partner system, the security level of the partner system can be modified with the command MODIFY-FT-PARTNER to meet the new requirements.

You can also use the operand SECURITY-LEVEL=*BY-PARTNER-ATTRIBUTES to activate the following automatic mechanisms for the security levels:

- Partners that are authenticated by *open*FT are assigned security level 10.
- Partners that are known in BCAM (i.e. they are addressed via their BCAM name) are assigned security level 90.
- Partners which are accessed via their IP address (only possible in the case of FTP) are assigned security level 100.

This automatic mechanism can be activated on a partner-specific basis (ADD-FT-PARTNER and MODIFY-FT-PARTNER) or globally by means of MODIFY-FT-OPTIONS.

If the security level is not specified at the partner system or if the partner system is not defined in the partner list (dynamic partner) then *open*FT uses the global settings in the operating parameters (MODIFY-FT-OPTIONS). Here, it is also possible to specify a fixed security level as the default.

3.4.1.3 Automatic deactivation

The MODIFY-FT-PARTNER command can be used to assign an “automatic deactivation” attribute to *open*FT partner systems. Automatic deactivation disconnects the partner after five failed attempts to establish a link. This prevents unnecessary costs from arising in the case of certain link types, such as DATEX-P, which also charge for unsuccessful link establishment attempts. Automatic deactivation does not work when the attempt fails due to an error in the BCIN commands in the local system, but neither are any costs incurred in such a case. Before any new attempts are made, the system must be manually reactivated.

3.4.2 Dynamic partners

Users may, as required, specify partners via their names in the partner list or via their addresses. In this way, they can also address partners that are not entered in the partner list.

Dynamic partners are partners which are either not entered in the partner list or which are entered in the partner list without a name (PARTNER-NAME=*NONE in the ADD-FT-PARTNER command). Hence, a user can specify a dynamic partner only via its address.

If a partner is not entered in the partner list then the same automatic mechanism applies for the security level as described for SECURITY-LEVEL=*BY-PARTNER-ATTRIBUTES, see page 42.

As FT administrator, you may also lock the use of dynamic partners for security reasons. You do this by means of the MODIFY-FT-OPTIONS command with the operand DYNAMIC-PARTNERS=*OFF. In this case, it is necessary to address partners via their names in the partner list. They cannot be addressed directly via their address. Inbound access is then also only permitted to partners that are entered in the partner list.

3.4.3 Backing up the partner list

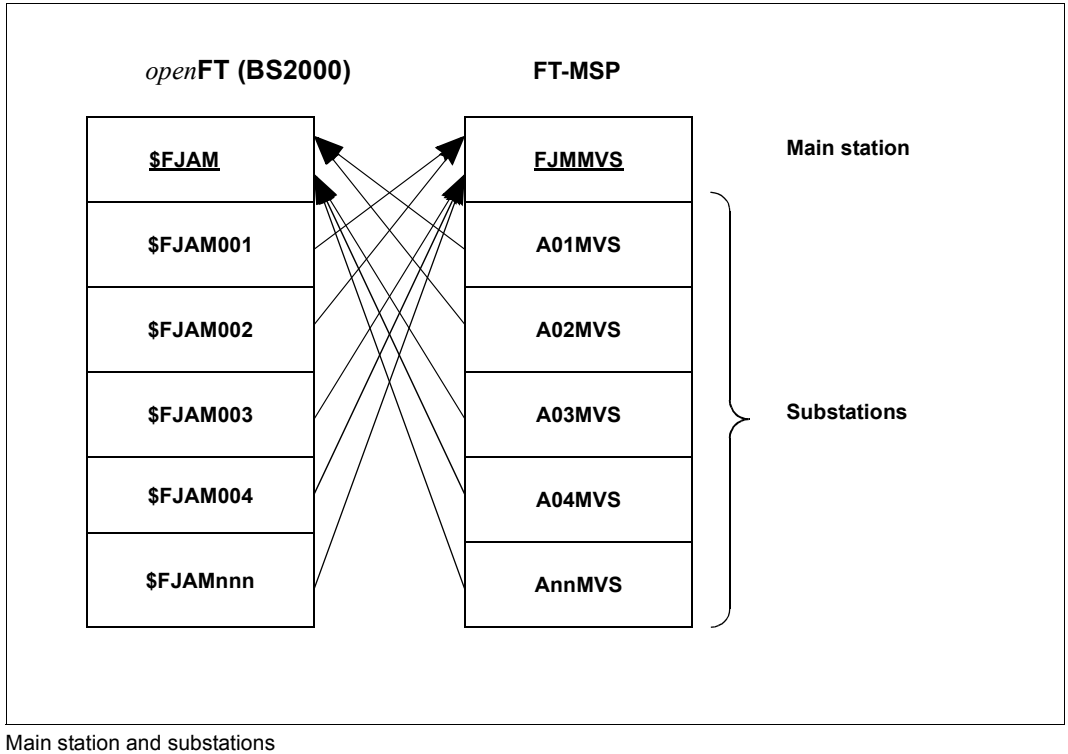
You can back up the entries in the partner list by means of the SHOW-FT-PARTNERS command or the START-OPENFTPART command:

- SHOW-FT-PARTNERS outputs the partner entries in the form of MODIFY-FT-PARTNER commands. To do this, specify the OUTPUT(LAYOUT=*BS2PROC) operand. The output can be redirected to a file by means of the SYSFILE command. To make the procedure executable, the first column of the SYSFILE output must then be removed in an editor.
- SHOW-OPENFTPART also outputs the partner entries to a file in the form of MODIFY-FT-PARTNER commands with the difference that the first column is already removed.

3.4.4 Addressing concept for partners up to *open*FT V8.0

In some transport systems, it is not possible to have multiple and concurrent transport connections between the same end points. In order to handle multiple FT requests simultaneously between two FT systems, all Fujitsu Siemens Computers FT products up to V8.0 have a standardized addressing concept which is described below. As of *open*FT V9.0 for BS2000 or z/OS and *open*FT V8.1 for UNIX systems or Windows systems, this addressing concept is superseded by network-wide, unique instance IDs for the *open*FT instances (see section “Using openFT in a HIPLEX cluster” on page 71). Compatibility is also provided for, so that you can link new and old versions together based on the previous addressing concept, without any problems.

In the traditional addressing concept, there is a so-called main station (main application) in each *open*FT system which serves as the end point for all links that are initiated in partner systems. The starting points for links to partner systems are the so-called substations (subapplications) in the local system.
The following diagram illustrates this principle, using the coupling of *open*FT for BS2000 and *open*FT for z/OS and OS/390 as an example:



Main station and substations

The main station of the partner system is entered in the partner list with the aid of the ADD-FT-PARTNER command (default value: $FJAM).

If *open*FT is used with the extended authentication check facility for *open*FT partners (PARTNER-CHECK=*TRANSPORT in the MODIFY-FT-OPTIONS and ADD-FT-PARTNER commands), the transport address of the partner is also checked against the entry in the partner list. The originator, however, is always one of the substations of an FT system, although it is the main station of the partner system that is entered in the partner list. This is the reason why naming conventions must be applied to ensure correct assignment. *open*FT partners that do not adhere to these naming conventions are rejected when extended authentication checking is in operation.

*open*FT recognizes two naming conventions:

1. If “$FJAM” was entered for the main station of the remote system (default value for BS2000 partners and computer interconnection with SINIX partners), any substation specification in the form “$FJAM*nnn*” is accepted, where *nnn* may be any three-digit number. The main station of *open*FT for BS2000 is always “$FJAM” and the associated substations are designated as described above without additional work being required at generation.
2. FJM*ftid* is entered as the main station of the partner system. *ftid* is a five-character alphanumeric string and must be unambiguous throughout the network.
   The substations are designated A*nnftid*, *nn* being two-digit numbers. *ftid* has the same meaning as for the main station. *ftid* must be the same for the main station and all its associated substations.

3.5 Security in FT operation

A user wanting to access resources of a system must always provide the system with proof of his or her authorization for the access. In the case of file transfer activities, access authorization must be verified in both the local and the remote system. Verification usually entails specifying a user ID and a corresponding password. Protection can be extended to the system-specific level by means of file passwords.

An even higher level of security in file transfer is offered by the product FTAC-BS2000 (File Transfer Access Control). FTAC provides the functions for controlling FT activities on a computer-specific and user-specific basis. The FTAC administrator defines authorization profiles for each user ID and if FTAC is in use, the FT administrator must assign security levels to the partner systems (see ADD-FT-PARTNER and MODIFY-FT-PARTNER SECURITY-LEVEL operand and section “FTAC security levels for partner entries” on page 42).

WARNING!

Note that *open*FT-AC is only effective for connected products such as *open*FT. If other file transfer products without an *open*FT-AC connection are also being used, a more comprehensive and coordinated security concept would be advisable.

3.5.1 Authentication

If data requiring a high degree of security is to be transferred, it is important to subject the respective partner system to a reliable identity check (“authentication”). The two *open*FT instances taking part in a transfer must be able to cryptographically check one another to determine whether they are connected to the “correct” partner instance.

Therefore, as of versions *open*FT V8.1 for UNIX systems and Windows systems, and V9.0 for BS2000 and z/OS, an expanded addressing and authentication concept is supported for *open*FT partners. It is based on the addressing of *open*FT instances using a network-wide, unique ID and exchanging partner-specific key information.

You should note that authentication in *open*FT for BS2000 is only possible for partners that are entered in the partner list!

When communicating with partners that are still using *open*FT version 8.0 (or earlier), the functions described in the following are not usable. For the sake of compatibility, the previous addressing concept is still supported for these partners. For FTAM partners, authentication is not available in this form, since the FTAM protocol standardized by the ISO does not provide for comparable functionality.

Instance identification

Each *open*FT instance that works with authentication must be assigned a network-wide, unique instance ID. This ID replaces the previous addressing of *open*FT instances using processor and application names. The instance ID is a unique name, up to 64 characters long. Its uniqueness must be based on something other than case-sensitivity. An instance ID may be comprised of alphanumeric characters and the special characters. You are advised only to use the special characters ".", "-", ":" or "%". The first character must be alphanumeric or the special character "%". The "%" character may only be used as a first character. An alphanumeric character must follow a "." character.

In order to ensure the network-wide uniqueness of instance IDs, you should proceed as follows when assigning them:

– If the *open*FT instance has a network address with a **DNS name**, you should use this as the ID. You can create an "artificial" DNS name for an openFT instance, by placing part of a name, separated by a period, in front of an existing "neighboring" DNS name.

– If the *open*FT instance does not have a DNS name, but is connected to a TCP/IP network, you should use the ID **%ipn.n.n.n** (n.n.n.n is the IP address of the local *open*FT instance, minus the leading zeros in the address components).

– If the *open*FT instance is connected to an ISDN network rather than a TCP/IP network, you should use the ID **%isdnmmmmmmmm** (mmmmmmmm is the ISDN call number, including country and local prefixes).

– If the *open*FT instance is connected to an X.25 network (but not to TCP/IP or ISDN), the ID should begin with **%x25** and the X.25 number should contain the NSAP, where necessary (e.g. **%x25mmmmmmmmmmNSAP)**.

You allocate these IDs for your local *open*FT instances using the IDENTIFICATION parameter of the /MODIFY-FT-OPTIONS command.

The form of instance ID used internally by *open*FT for partners using a version earlier than V8.1, (i.e. **%.<prozessor>.<entity>)**, should not be used explicitly for partners.

Store instance IDs of partner systems in the partner list using the IDENTIFICATION parameter of the ADD-FT-PARTNER command, or MODIFY-FT-PARTNER. With the aid of the partner systems' instance IDs, *open*FT manages the resources assigned to those partners, such as request hold queues and cryptographic keys.

Creating and managing local keys

In order to ensure that your own *open*FT instance can be authenticated in the partner system, a suitable public key for the instance must be made available to the partner system. Using CREATE-FT-KEY-SET, you can create an RSA key pair set for the local *open*FT instance; this consists of a private (secret) key and a public key.
Private keys are internally administered by *open*FT. Public keys are stored on the configuration user ID of the openFT instance (standard: $SYSFJAM), under the name **SYSPKF.R<key reference>.L<key length>.** The key reference is a numeric designator for the version of the key pair, the default key length is currently 768 in BS2000. The public key files are text files, which are created in the character code of the respective operating system, i.e. EBCDIC.DF04-1 for BS2000 and z/OS, ISO8859-1 for UNIX systems and CP1252 for Windows systems.

In a SYSPKF.COMMENT file on the configuration user ID of the *open*FT instance, you can store comments, which are written in the first lines of the public key files when a key pair set is created. Comments could, for example, contain the contact data for the FT administrator on duty, the computer name, or similar information that is important for partners. The lines in the SYSPKF.COMMENT file may be a maximum of 78 characters in length. Using the UPDATE-FT-PUBLIC-KEYS command, you can import updated comments from this file into existing public key files at a later time.

If a public key file has been unintentionally deleted or otherwise manipulated, you can re-create the public key files of the existing key pair sets using UPDATE-FT-PUBLIC-KEYS.

If you want to replace a key pair set with a completely new one, you can create a new key pair set using CREATE-FT-KEY-SET. You can identify the most current public keys by the highest value key reference in the file name. *Open*FT supports a maximum of three key pair sets at a time. The existence of several keys should only be temporary, until you have made the most current public keys available to all the partner systems. Afterwards, you can delete the key pair sets no longer needed using DELETE-FT-KEY-SET.

If the *open*FT administrator is not the same as the system administrator, it must be ensured that this administrator has access to the SYSPKF files and the SYSKEY library on the configuration user ID of the *open*FT instance. This can be done, either by assigning operating system-specific access rights or by setting up corresponding FTAC admissions profiles.

Distributing the keys to partner systems

Distributing the public key files to your partner systems should take place by secure means, for example by

– distribution by cryptographically secure e-mail

– distribution on a CD (by courier or by registered mail)

– distribution via a central *open*FT file server, the public keys of which are in the partners' possession

If you transmit your public key files to partner systems using UNIX or Windows operating system, you must ensure that these files are re-coded from EBCDIC.DF04-1 to ISO 8859-1 or CP1252 (e.g. by transferring them as a text file via *open*FT).

The public key file of your local *open*FT instance is stored in the partner system in the following location:

– For partners with *open*FT for BS2000, as a type D PLAM element in the **SYSKEY** library, the configuration user ID of the partner instance. The partner name allocated for your *open*FT instance in the remote network description file or in the remote partner list must be selected as the element name.

– For partners with *open*FT for UNIX systems, in the **/var/openFT/<instance>/syskey** directory. The instance ID of your local *open*FT instance must be selected as the file name. The file name must not contain any uppercase letters. If the instance ID contains uppercase letters, these must be converted to lowercase in the file name.

– For partners with *open*FT for Windows, in the **openFT\var\<Instance>\syskey** directory. The instance ID of your local *open*FT instance must be selected as the file name.

– For partners with *open*FT for z/OS or OS/390, as a PO element in the **<admuser>.SYSKEY** library. The partner name allocated for your openFT instance in the remote network description file or partner list must be selected as the element name.

Managing the keys of partner systems

The public keys of the partner systems are stored in BS2000 as type D PLAM elements in the **SYSKEY** library on the configuration user ID of the local *open*FT instance. The partner name of the partner system as defined in the partner list must be selected as the element name. If an updated public key is made available by the partner instance, the old key must be overwritten by it.

Local and partner authentication

There are three distinct usages:

– The local *open*FT instance checks the identity of the partner instance. This assumes that a current, public key of the partner instance was stored locally in the SYSKEY library, the name of which corresponds to the partner name of the partner instance defined in the partner list. This sort of configuration makes sense, for example, if files on a file server are to be accessed using *open*FT. It is important for the local *open*FT instance, that the retrieved data should come from a reliable source (from the authenticated partner). In contrast, the file server is not concerned with who is accessing it.

– The partner instance checks the identity of the local *open*FT instance. This assumes that a current, public key of the local *open*FT instance is stored in the partner instance (re-coded - for UNIX and Windows partners). This sort of configuration would be considered, for example, if partner systems in several branch systems are to be accessed using *open*FT from a central computer, and where the branch system computers only allow the central computer access (and, in practice, only the central computer).

– Both of the *open*FT instances engaged in a transfer authenticate each other. This assumes that current, public keys were mutually exchanged and the partners are addressing each other using their instance IDs. In this way, it can be ensured that the data not only comes from a reliable source, but that it will also end up in reliable hands.

3.5.2 Extended authentication check

*open*FT partners using *open*FT from version 8.1 onwards, support the authentication mechanism (see page 46). If the local system has a public key of the partner at its disposal, the partner's identity is checked by cryptographic means.

For partner systems that do not work with authentication, inbound requests are checked with the aid of the processor name, in order to ascertain whether the calling system has a valid entry in the partner list. As an option, *open*FT offers the possibility of checking (via sender checking) not only the processor name, but also the transport address.

The extended sender checking can be globally enabled for *open*FT partners or just for specific partners:

- globally, using
  MODIFY-FT-OPTIONS... PARTNER-CHECK=*TRANSPORT-ADDRESS
- only for specific partners, using
  ADD-FT-PARTNER ... PARTNER-CHECK=*TRANSPORT-ADDRESS or
  MODIFY-FT-PARTNER ... PARTNER-CHECK=*TRANSPORT-ADDRESS

The global setting is valid for all partners with the value PARTNER-CHECK=*BY-FT-OPTIONS (default in the ADD-FT-PARTNER).

For FTAM partners, expanded sender checking is essentially enabled. Therefore, this characteristic can neither be displayed nor processed for FTAM partners.

Use of the extended authentication check while dynamic partners are also enabled is only of value if FTAC functionality is being used. Even if the extended authentication check is enabled, it will not be evaluated if dynamic partners are also enabled.

If the authentication check returns a negative result, the request is rejected.

3.5.3 Protection mechanisms for file transfer

*open*FT supports for *open*FT partners the encryption of the data sent and received in the process of setting up the connection and processing a file transfer request. The partners involved in file transfer automatically negotiate encryption and use of the appropriate public key in the process of connection set-up. *open*FT automatically encrypts the request description data provided that the partner supports this functionality.

Using the CREATE-FT-KEY-SET command, the FT administrator must create at least one key pair set, upon which the encryption will be based and carried out.

If the file content is to be encrypted for transfer by *open*FT, the optional *open*FT-CR component must be installed.

The encrypted transfer of file contents is requested with the TRANSFER-FILE command and only executed if *open*FT is installed with *open*FT-CR in both the systems involved in the file transfer. If one of the two systems is not capable of handling encrypted file transfers, the request is rejected with the message FTR2111 (no *open*FT-CR in local system) or with FTR2051 (no encryption possible in the remote system).

For legal reasons, *open*FT-CR is not available in all countries.

In BS2000, if the *open*CRYPT subsystem is installed and started in addition to *open*FT-CR, then *open*FT itself does not encrypt the file content, but allows *open*CRYPT to handle the encryption. This considerably enhances performance.

3.5.4 Protection mechanisms to ensure file consistency

Prior to version V8.0, FT products in BS2000 protected a file to be transferred only during the active transmission, i.e., when the file was opened by *open*FT using DVS. Consequently, if the transmission was interrupted or even if the transmission had not yet begun, both files involved could be potentially accessed and modified. Such changes could not always be detected on restarting *open*FT, thus resulting in the creation of inconsistent receive files.

*As of V8.0, open*FT uses an operating system mechanism to protect transfer files (however, this protection is not possible for library elements and Posix files):

– When a file transfer request is accepted, a lock is set on each file to be transferred as early as possible. Only read access is granted to other users for the send files; no access is permitted for the receive files.

– This lock remains set - so long as the FT subsystem is loaded - until the request has completed.

– The BS2000 command SHOW-FILE-LOCK indicates whether a file has been locked by *open*FT and, if it is, shows the transfer ID (or, when sending, possibly a list of transfer IDs) of the request involved. Such locks, and other file locks as well, can be reset by the system administrator at his/her own discretion in emergency situations by using the command REMOVE-FILE-ALLOCATION.

– Using SHOW-FILE-TRANSFER ... PUBSET=, the FT administrator can have all the requests displayed that have locked files on a defined pubset. The administrator can selectively delete these requests using CANCEL-FILE-TRANSFER ... PUBSET=.

On unloading an FT instance (STOP-SUBSYSTEM FT or DELETE-FT-INSTANCE), all the locks held by *open*FT are cleared and reset upon reload (START-SUBSYSTEM FT or CREATE-FT-INSTANCE) for all files affected by existing requests. The FT or system administrator must therefore observe the following:

– On loading an *open*FT instance, all pubsets which contain files that may need to be transferred by *open*FT must be available, since *open*FT will otherwise be unable to set the file lock, thus causing the respective transmission to be aborted.

– The loading must be completed early enough to ensure that the files to be transmitted are protected in time.

– An *open*FT instance should be unloaded as late as possible, but before exporting the pubsets, on which the transfer files are located.

In addition to this mechanism, *open*FT also implicitly checks the integrity of the transferred data by communicating with *open*FT partners version V8.1 and later. The scope is defined in the transfer request:

– In the case of requests with encryption, the transferred file content is also checked.

– In the case of requests without encryption, an integrity check of the file content can be activated explicitly.

– If neither encryption nor the integrity check are activated then only the integrity of the request description data is checked.

If an error is detected then restartable requests attempt the transfer again. Requests that cannot restart are aborted.

3.6 Monitoring and controlling FT operation

Fetch information on the FT system

The FT administrator uses the following commands to obtain information on the system:

SHOW-FT-OPTIONS	Information on operating parameters
SHOW-FT-PARTNERS	Information on partner systems
SHOW-FT-LOGGING-RECORDS	Information on log entries
SHOW-FILE-TRANSFER	Information on file transfer status
SHOW-FT-INSTANCE	Information on *open*FT instances

The SHOW-FT-OPTIONS command furnishes information on the current settings of the operating parameters.

SHOW-FT-PARTNERS yields information on the partner systems and their associated properties, e.g., names, addresses, security levels for FTAC, and so on. The command and the possible outputs are described in detail starting on page 271.

To support automatic monitoring, some events which are not direct responses to user input are reported by *open*FT via console messages. More detailed information on this topic can be found in the section “Console messages for automatic monitoring” on page 56.

The command SHOW-FT-LOGGING-RECORDS can be used to display the logs of file transfer requests. You will find more information on this subject in the section below and in the description of the SHOW-FT-LOGGING-RECORDS command on page 249ff.

SHOW-FILE-TRANSFER enables the FT administrator to retrieve information on all file transfer requests in his or her system, even when the FT system is stopped.

Using SHOW-FT-INSTANCE, the FT administrator can find out which *open*FT instances exist in the system and have their characteristics and status displayed.

3.6.1 FT logging

The following 3 commands are available for the FT logging function:

DELETE-FT-LOGGING-RECORDS	Delete logging records
MODIFY-FT-OPTIONS	Switch on/off logging function
SHOW-FT-LOGGING-RECORDS	View information on log entries

*open*FT records the results of all file transfer requests, irrespective of whether the initiative is in the local or the remote system (outbound and inbound requests, respectively). The information on each successfully completed or aborted request is recorded in an FT logging record. The file consisting of these logging records thus represents a complete, uninterrupted documentary record of FT operation over a prolonged period of time.

FT logging can be activated and deactivated at any time by means of MODIFY-FT-OPTIONS.

*open*FT writes the logging records into the log file SYSLOG on the configuration user ID of the *open*FT instance (default: $SYSFJAM). The SYSLOG file is created by the FT system with second allocation 500, its net size depends on the number of logging records it contains. As one of your duties as FT administrator, you should regularly create backups of logging-records (the frequency depends on the volume) as a printout, a file in CSV format or on tape, for example and then delete the logging records with the DELETE-FT-LOGGING-RECORDS command. In this way you have a complete, uninterrupted log at your disposal for documentation purposes, while at the same time no storage capacity is wasted. Bear in mind the assigned file size does not change, but the space formerly occupied by the records you delete is released within the file.

You can find further information on the Internet under *www.fujitsu-siemens.de/openft* - application scenarios.

The information content of the FT logging records includes:

– date and time of request processing,
– an acknowledgment indicating correct completion of a request, or the reason for request rejection or abort,
– the direction of file transfer,
– the name of the partner system involved in file transfer.
– TSN and user ID of the request initiator for requests submitted in the local system; only *REMOTE is entered for remote request initiators,
– the user ID under which the request was handled or should have been handled,
– the name of the file.
– if an abort occurs, additional information on the cause.

The FT administrator can use the SHOW-FT-LOGGING-RECORDS command to output all FT logging records of his/her system to SYSOUT or SYSLST. Two formats are available for the output: a format that is suitable for listings, and a format that is optimized for further

processing. The FT administrator can also choose between a short form and a long form, i.e., a brief overview or a long detailed output. If the FTAC functionality is being used, the logging records relevant for FTAC are saved in the same file. A detailed description of the command SHOW-FT-LOGGING-RECORDS can be found on page 249ff; the output is presented starting on page 256.

3.6.2 Console messages for automatic monitoring

Messages are usually issued as responses to administration commands. There are, however, also some messages which are not (or not exclusively) issued by administration commands. These messages are described in the User Guide “Messages”. When errors occur on accessing the request queue or the partner list, *open*FT generates normal DMS error messages.

To support automatic monitoring, some events which are not direct responses to user input are reported by *open*FT via a console message. Depending on which events are involved, further actions can then be initiated by automatic operators such as Omnis-Prop, HLL-Prop, etc. Console messages can also be used to generate SNMP traps for automatic FT monitoring using SNMP.

The console messages for automatic monitoring occupy the message code range from FTR0300 to FTR0399. They have the routing code ‘@’, which means that they must be explicitly requested, for example, using the following command

```
/MOD-MSG-SUBSCRIPTION ADD-MSG-ID=(FTR0301,FTR0307,FTR0340,FTR0341)
```

Messages for monitoring partner systems

FTR0301 Partner '(&00)' entered state NOCON

FTR0302 Partner '(&00)' entered state ACTIVE

FTR0303 Partner '(&00)' entered state LUNK

FTR0304 Partner '(&00)' entered state RUNK

FTR0305 Partner '(&00)' entered state INACTIVE

FTR0306 Partner '(&00)' entered state AINAC

FTR0307 Partner '(&00)' may be unreachable

FTR0308 Partner '(&00)' does not allow any more inbound requests

FTR0309 Partner '(&00)' added

FTR0310 Partner '(&00)' removed

FTR0311 Partner '(&00)') entered state LAUTH

FTR0312 Partner '(&00)' entered state RAUTH

FTR0313 Partner '(&00)' entered state DIERR

FTR0314 Partner '(&00)' entered state NOKEY

FTR0315 Partner '(&00)' entered state IDREJ

Messages for monitoring *open*FT

FTR0320 abnormal termination initiated

FTR0360 openFT control process started

FTR0361 openFT control process terminated

Messages for monitoring the request queue

FTR0330 Request queue 85 percent full

FTR0331 At least 20 percent of request queue unoccupied

Messages for monitoring requests

FTR0340 Transfer '(&00)' successfully completed

FTR0341 Transfer '(&00)' terminated with error

3.6.3 SNMP management for *open*FT

SNMP stands for **S**imple **N**etwork **M**anagement **P**rotocol and was developed as the protocol for network management services in TCP/IP networks. *open*FT permits you to centrally monitor and administer one or more *open*FT systems from one central management station using graphical interfaces. A prerequisite for SNMP-based *open*FT management is the installation of the products SNMP Management ≥ V6.0, SNMP Basic Agent BS2000 V6.0 (SBA-BS2) and SNMP Standard Collection BS2000 V6.0 (SSC-BS2).

Detailed information can be found in the respective user manuals.

To support automatic monitoring, some events which are not direct responses to user input are reported by *open*FT via a console message. Console messages can also be used to generate SNMP traps for automatic FT monitoring using SNMP.

If the file transfer subagent is used then *open*FT itself can also generate SNMP traps (without having to use console messages).

The file transfer subagent is used to:
– start and stop *open*FT for BS2000
– acquire system parameter information
– change the public key for encryption
– output statistics
– diagnostic control
– output partner information

The proprietary MIB for *open*FT contains objects for the management tasks listed above. The objects for starting and stopping, changing the public key for encryption, and for diagnostic control also provide write access.

3.6.3.1 Starting and stopping *open*FT

MIB definition	Access	Meaning
ftStartandStop	read-write	Start / Stop

*open*FT is started and stopped via the *open*FT subagents by setting the value “START” or “STOP” respectively. A read access returns information on the current FT system state.

3.6.3.2 System parameters

MIB definition	Access	Meaning
ftSysparVersion	read-only	Version
ftSysparTransportUnitSize	read-write	Transport Unit Size
ftSysparTaskLimit	read-write	Task Limit
ftSysparConnectionLimit	read-write	Connection Limit: maximum number of transport connections that can be reserved for the execution of FT requests
ftSysparPartnerCheck	read-write	Partner Check
ftSysparMaxInboundRequests	read-write	Max Inbound Requests: maximum number of inbound requests per partner system
ftSysparMaxLifeTime	read-write	Max Request Lifetime: maximum lifetime (in days) in the request queue

Further information of the output values can be found in the section on the SHOW-FT-OPTIONS command on page 263.

3.6.3.3 Public key for encryption

MIB definition	Access	Meaning
ftEncryptKey	write-only	An entry of “create-new-key” or “1” causes a new public key to be generated.

3.6.3.4 Statistics

MIB definition	Access	Meaning
ftStatSuspend	read-only	Requests in a SUSPEND state
ftStatLocked	read-only	Requests in a LOCKED state
ftStatWait	read-only	Requests in a WAIT state
ftStatActive	read-only	Requests in an ACTIVE state
ftStatFinished	read-only	Requests in a FINISHED state
ftStatCanceled	read-only	Requests in a CANCELD state
ftStatHold	read-only	Requests in a HOLD state
ftStatLocalReqs	read-only	Async requests in the local system
ftStatRemoteReqs	read-only	Requests in the remote system

A description of the output values can be found in the section on the SHOW-FILE-TRANSFER command on page 224.

3.6.3.5 Diagnostic control

MIB definition	Access	Meaning
ftDiagStatus	read-write	on / off
ftDiagFtamPartners	read-write	on / off
ftDiagOpenftPartners	read-write	on / off
ftDiagFtpPartners	read-write	on / off
ftDiagSynRequests	read-write	on / off
ftDiagAsynRequests	read-write	on / off
ftDiagLocRequests	read-write	on / off
ftDiagRemRequests	read-write	on / off
ftDiagOptionsNobulk	read-write	on / off

Please also read the section on the MODIFY-FT-OPTIONS command on page 178.

3.6.3.6 Partner Information

MIB definition	Access	Meaning
ftPartnerName	read-only	Name of the FT partner
ftPartnerType	read-only	FT protocol used by the partner
ftPartnerState	read-write	Status of the FT partner: act (1), inact (2), nocon (3), lunk (4), runk (5), adeact (6), ainact (7) lauth (8) rauth (9) dierr (10) nokey (11) idrej (12)
ftPartnerAddress	read-only	Address of the partner system

Only a status update for one partner is supported at present, and only the values act, inact and adeact may be specified.

3.6.3.7 Traps

Object name	Trap No.	Explanation
Enterprise = sniFTTraps		
ftStopTrap	1	TRAP is sent if *open*FT is terminated
ftPartnerStateTrap	4	TRAP is sent if the partner status has changed
ftPartnerUnreachableTrap	5	May not be possible to access partner
ftStartTrap	6	TRAP is sent after start of *open*FT
ftRequestQueueUpperLimitTrap	7	TRAP is sent if the FT request queue is more than 85% full
ftRequestQueueLowerLimitTrap	8	TRAP is sent if at least 20% of the FT request queue is free again
ftRequestSuccessfulTrap	9	TRAP is sent if an FT request is sent successfully
ftRequestErrorTrap	10	TRAP is sent if an FT request is terminated with an error
ftSubsystemStartTrap	11	TRAP is sent if the FT subsystem has been started
ftSubsystemStopTrap	12	TRAP is sent if the FT subsystem has been stopped

3.6.3.8 Trap groups and trap controls

The traps of the *open*FT subagent can be gathered together into groups that are represented by the following MIB objects. This means that you can enable or disable the sending of traps for the individual trap groups as follows (trap send status “on” or “off”):

– Specification 2 (“on”): the traps for the group in question are sent.
– Specification 1 (“off”): the traps for the group in question are not sent.

MIB definition	Access	Affected traps
ftTrapsSubsystemState	read-write	– ftSubsystemStartTrap – ftSubsystemStopTrap
ftTrapsFTState	read-write	– ftStartTrap – ftStopTrap
ftTrapsPartState	read-write	– ftPartnerStateTrap
ftTrapsPartnerUnreachable	read-write	– ftPartnerUnreachableTrap
ftTrapsRequestQueueState	read-write	– ftRequestQueueUpperLimitTrap – ftRequestQueueLowerLimitTrap
ftTrapsTransSucc	read-write	– ftRequestSuccessfulTrap
ftTrapsTransFail	read-write	– ftRequestErrorTrap

3.6.3.9 Trap information

The MIB of the *open*FT subagent contains definitions of MIB objects which are sent together with the traps.

MIB definition	Access	Explanation
ftRequestID	not-accessible	Transfer ID of the request
ftRequestInitiator	not-accessible	Initiator of the request: local (1), remote (2)
ftRequestPartnerName	not-accessible	Partner
ftRequestUserID	not-accessible	User ID of submitter
ftRequestFileName	not-accessible	Name of the file for transfer
ftRequestError	not-accessible	Error in request

3.7 Administrating and controlling FTAC functions

3.7.1 Creating a default admission set

The FTAC administrator must first determine an average protection level for the user IDs in his system and use this information to modify the default admission set, whose values after the installation of *open*FT-AC are all 0. In the default admission set, the settings are made for the “average” FTAC user in the system. This provides adequate protection for most users. These specifications are valid for all user IDs which do not have their own admission set. Furthermore, in each admission set, the entry *STD can be used in different places to refer to the default admission set. This has the advantage of automatically incorporating any modification of the default admission set into these admission sets.

The FTAC administrator can set individual values for user IDs whose protection requirements deviate from the average.

3.7.2 Administrating admission sets

For the administration of admission sets, *open*FT-AC offers the FTAC administrator the following commands:

MODIFY-FT-ADMISSION-SET	Modify admission sets
SHOW-FT-ADMISSION-SET	Show admission sets

A maximum security level is specified in the admission set for each of the six basic functions. The user ID with this admission set can use this basic function with all partner systems who have this security level or lower.

The FTAC administrator can view admission sets with the command SHOW-FT-ADMISSION-SET (see page 244). He can modify the admission sets with the command MODIFY-FT-ADMISSION-SET (see page 169).This command is used to modify the default admission set as well as to customize the settings for individual user IDs. The specifications of the FTAC administrator are the MAX-ADM-LEVELS in the admission set for the corresponding user ID. The user can increase the degree of protection within these levels (MAX-USER-LEVELS).

3.7.3 Administrating admission profiles

For the administration of admission profiles, *open*FT-AC offers the FTAC administrator the following commands:

CREATE-FT-PROFILE	create admission profile
DELETE-FT-PROFILE	delete admission profile
MODIFY-FT-PROFILE	modify admission profile
SHOW-FT-PROFILE	show admission profile

The FTAC administrator has the option of modifying foreign admission profiles:

– He can view them with the command SHOW-FT-PROFILE (see page 279). The transfer admission of an admission profile is not output. This means that the FTAC administrator does not have access rights to the files of foreign user IDs.

– He can delete them with the command DELETE-FT-PROFILE (see page 156). This is the most radical of all options which should only be used in extreme cases and with good reason and upon consultation with the owner of the profile.

– He can privilege them with the command MODIFY-FT-PROFILE (see page 199), or conversely revoke privileges.

– He can also modify them with MODIFY-FT-PROFILE. Access to the admission profile will then be blocked. If the FTAC administrator also possesses the TSOS privilege then the profiles are not locked.

Privileging admission profiles

In exceptional cases, the FT user can use a privileged admission profile to disregard the specifications of own admission profile. The user ID protection is maintained in this case, by the fact that only very restricted access is permitted into the admission profile. Exceptional cases where this is allowed include:

– if a particular file needs to be transferred,

– if follow-up processing is not permitted or severely restricted,

– if a partner system with a higher security level is permitted to carry out file transfers with the user ID, but others with lower security levels are not.

The procedure to follow when privileging an admission profile is simple:

1. The user creates an admission profile for the planned task with the command CREATE-FT-PROFILE.
2. The FTAC administrator views the admission profile with the command SHOW-FT-PROFILE to determine if the profile presents a threat to data security.

   Example

   ```
   /SHOW-FT-PROFILE NAME=PROFPROD,
                    SELECT-PARAMETER=(OWNER-IDENTIFICATION=DONALD),
                    INFORMATION=*ALL
   ```

   Short form:

   ```
   /SHOW-FT-PROF PROFPROD,SEL=(,DONALD),INF=*ALL
   ```

   The output has the following form:

   ```
   %PROFPROD
   % IGN-MAX-LEV = (IBR)
   % FILE-NAME   = PROFIT
   % USER-ADM    = (DONALD,M4711DON,OWN)
   % PROC-ADM    = SAME
   ```

   The first line of the output shows the name of the admission profile, the second line the values which Donald has set in the command CREATE-FT-PROFILE (page 127) or which are determined by the default values, if Donald doesn't set them himself.

3. If the profile will not endanger security, the FTAC administrator privileges it with the help of the command MODIFY-FT-PROFILE.

   Example

   ```
   /MODIFY-FT-PROFILE NAME=PROFPROD,
                      SELECT-PARAMETER=(OWNER-IDENTIFICATION=DONALD),
                      PRIVILEGED=*YES
   ```

In a privileged admission profile, only the transfer admission and the parameter PRIVILEGED may be modified. This prevents the misuse of any profiles, once privileged.

3.7.4 Transfer FTAC environment - the environment functions

The following commands are available for the environment functions:

EXPORT-FTAC-ENVIRONMENT	output FTAC environment to file
IMPORT-FTAC-ENVIRONMENT	transfer FTAC environment from file
SHOW-FTAC-ENVIRONMENT	show FTAC environment from export file

The FTAC administrator can have admission profiles and sets written (i.e. “exported”) to a file and thus back up all admission profiles and sets that exist on the computer. In addition, this function is useful when a user migrates from one computer to another. In this case, the FTAC administrator first backs up the existing FTAC environment to a file and then re-installs this on another computer. The FTAC user can then continue to work in the same FTAC environment as before, i.e. with the same admission profiles and the same admission set. Any existing privileges must be explicitly set up again on the new computer, and the admission profiles must be explicitly released by their owners if the FTAC administrator does not possess the TSOS privilege. On the other hand, if the FTAC administrator has the TSOS privilege, he/she can specify on importing whether the profiles will be imported with unmodified attributes or not.

The FTAC administrator can also selectively back up (EXPORT-FTAC-ENVIRONMENT, page 159) admission sets and profiles by using corresponding parameter specifications and then restore them when needed (IMPORT-FTAC-ENVIRONMENT, page 161). This can be done with:

– admission profiles and admission sets of one or more users (up to 100)
– all admission profiles and admission sets on a given computer
– only admission sets, no admission profiles
– only admission profiles, no admission sets

The contents of a backup file can be viewed with the command:

SHOW-FTAC-ENVIRONMENT

which displays the FTAC environment from the export file (see page 240).

Example

Donald Duck needs to work on a new computer under the same user ID DONALD. Donald would like to keep the same admission set and admission profiles as before. To do this, the FTAC administrator Dagobert backs up the admission set and the admission profiles for the user ID DONALD in the file DONALD.FTAC.BKUP.

```
/EXPORT-FTAC-ENVIRONMENT TO-FILE=DONALD.FTAC.BKUP,
                         USER-IDENTIFICATION=DONALD
```

Being a conscientious FTAC administrator, Dagobert Duck checks if the desired backup is in the file DONALD.FTAC.BKUP.

```
/SHOW-FTAC-ENVIRONMENT FROM-FILE=DONALD.FTAC.BKUP
```

He receives the following output:

```
              MAX. USER LEVELS               MAX. ADM LEVELS          ATTR
% USER-ID  OBS  OBR  IBS  IBR  IBP  IBF   OBS  OBR  IBS  IBR  IBP  IBF
% DONALD     1    1    0    1    0    0     1    1    0    0    0    0
% OWNER     NAME
% DONALD   *PROFPROD
```

Now, Dagobert transfers the file DONALD.FTAC.BKUP to the user ID of the FTAC administrator on the new computer.
There, Sylvester the Cat, the FTAC administrator for the new computer, transfers the admission set and the admission profiles of the user ID DONALD from the file DONALD.FTAC.BKUP.

Sylvester is also a conscientious administrator. He checks if Donald's admission sets and profiles are a threat to the security of his system (he doesn't trust Dagobert in the slightest):

```
/SHOW-FTAC-ENVIRONMENT FROM-FILE=DONALD.FTAC.BKUP
```

and of course he receives the same output as above.

Then Sylvester imports Donald's admissions from the file DONALD.FTAC.BKUP onto his system:

```
/IMPORT-FTAC-ENVIRONMENT FROM-FILE=DONALD.FTAC.BKUP
```

Since Sylvester the Cat does not possess the TSOS privilege, he must now privilege Donald's profile:

```
/MOD-FT-PRO UMSAWARE,,(,DONALD),PRIV=*Y
```

Finally, Donald must release the imported profiles before he can work with them. This would not be necessary if Sylvester the Cat possessed the TSOS privilege.

```
/MODIFY-FT-PROFILE NAME=*ALL,
                   TRANSFER-ADMISSION=*OLD(VALID=*YES)
```

3.7.5 The FTAC logging function

*open*FT-AC checks the access rights of every FT request which the protected system is involved in and logs the results. This information is stored in the so-called FTAC logging records.
The following information can be called up by the FTAC administrator:

- logging date
- type of logging record (FT or FTAC logging record)
- logging number of the FT request
- time of access check
- code for the function of the FT request (see table)
- reason for any rejections of the request by FTAC (the User Guide contains an overview of the codes for these reasons)
- transfer direction of the FT request
- name of the partner system with which the FT request was/is to be carried out
- TSN process sequential number and USER-IDENTIFICATION LOGON authorization of the initiator of requests which were made in the local system (or *REMOTE for remote request initiators)
- name and privileging identifier of any admission profiles used
- the local file or library name

*open*FT-AC checks the authorization of requests using the admission sets and profiles. *open*FT logs in the FT logging records whether this request can actually be carried out by *open*FT. Normally, there are two logging records per request. If FTAC rejects a request because of a negative access check, then there will be an *open*FTAC-AC logging record but no FT logging record. The display of *open*FT-AC logging records can not be turned off. However, the MODIFY-FT-OPTIONS command can be used to restrict it to requests rejected by FTAC (*REJECTED) or to modified requests.

The FT command SHOW-FT-LOGGING-RECORDS can be used by the FTAC administrator to find out about all access checks which have been carried out by *open*FT-AC to date (see page 249). This facilitates processes such as system inspections.

Codes for the function of the FT request

The entries in front of the brackets indicate the log representations of the individual FT functions. The FT requests themselves can consist of groups of FT functions. However, only one will appear in the logging record. These groups are listed in the brackets.

␣	TRANSFER-FILE	(WRITE-FILE + ... or READ-FILE + ...)
V	MOVE-FILE	(READ-FILE + DELETE-FILE)
A	READ-FILE-ATTRIBUTES	(READ-FILE-ATTRIBUTES + ...)
D	DELETE-FILE	(DELETE-FILE + ...)
C	CREATE-FILE	(CREATE-FILE + ...)
M	MODIFY-FILE-ATTRIBUTES	(MODIFY-FILE-ATTRIBUTES + ...)
R	READ-DIR	(READ-DIR + ...)
CD	CREATE-DIR	
MD	MODIFY-DIR	
DD	DELETE-DIR	

To make the output of the command SHOW-FT-LOGGING-RECORDS provide more of an overview, you can specify values or value ranges for various output parameters when calling up the command. This permits you to be selective in the output of logging records.

Deleting logging records

The FT administrator and the FTAC administrator are the only users in the system who can not only view but also delete the FTAC logging records. The corresponding FT command is DELETE-FT-LOGGING-RECORDS (see page 152). The FTAC user cannot use this function.

FTAC logging records can only be deleted from the oldest date up to a specified date. This ensures that there will be no gaps in the log file up to the most current record.

In theory, *open*FT-AC can write any number of logging records (“until the disk is full”). From time to time, the FTAC administrator should make a backup of existing logging records (either print out a hard copy or make a copy on tape or save a file in CSV format) and then delete these logging records from the log file. This ensures that the logging records will provide a continuous record over an extended period of time, as well as prevent the log file from getting too large.

For further information, refer to “Auditable Logging Archive” in the brochure “*open*FT - The Classic Solution for File Transfer and More” (suggested solution including procedures) or consult the application scenarios available on the Internet at www.fujitsu-siemens.com/openft.

3.8 Using *open*FT in a HIPLEX cluster

In *open*FT you can run multiple *open*FT instances on one computer simultaneously. Because of these instances, should a computer fail, you are in a position to carry over the functionality of the *open*FT to another computer, which is already running *open*FT.

After installing *open*FT, the **default instance** exists on each computer. This instance is atypical in that it cannot be deleted by instance management commands. Its application data is located on the default pubset under ID $SYSFJAM. When instances are displayed (SHOW-FT-INSTANCE), the default instance is always displayed first.

Up to 16 additional instances can be created by administration. Each of these instances, including the standard instance, consists of the following components:

- The request file SYSRQF, the partner list SYSPTF, the logging file SYSLOG, trace files, options SYSOPF and the profile file SYSFSA. Each instance therefore requires a configuration user ID with the characteristics that are described for the SYSFJAM ID (see section “Preparing the FT system” on page 18)
- Each instance requires its own network address; this always remains the same, independent of the real host. The host name must therefore be stored in the options using the MODIFY-FT-OPTIONS command. This (virtual) BCAM host must always be accessible under the same network address. In order to prevent the BCAM connection setup from automatically being passed to the real host instance when an instance fails to start, BCAM aliasing should be disabled for the $FAM and $FTAM applications.

The *open*FT installation files are only available once per computer and are shared by all the instances. The same version, however, must be installed on all the computers in the cluster (*open*FT version, proofing version, reps, etc.).

*open*FT commands that are called during a preprocessing, postprocessing or follow-up processing session, run under the same instance as the request that initiated the processing.

Commands for administrating *open*FT instances

As the *open*FT administrator, you can create, modify, and delete instances. In addition, you can set and get information on instances (like a user). The creation, modification and deletion of instances is only possible via the SDF interface, not via the POSIX command interface.

- Creating an instance

  Using the CREATE-FT-INSTANCE command, you can create an instance.

  If an instance is created, an entry is made in the administration file. This entry consists of the name of the instance and the pubset and user ID in which the files required for operation are stored (the request file, partner list, etc.). All the initialization tasks are carried out in the same way as during START-SUBSYSTEM. In the event you have also specified the AUTOMATIC-START option, then *open*FT is subsequently and immediately started in this instance.

- Modifying an instance

  With the MODIFY-FT-INSTANCE command, you can rename an instance and modify its AUTOMATIC-START characteristics.

- Deleting an instance

  With the DELETE-FT-INSTANCE command, you can delete an instance. Deleting an instance removes the administration entry for the instance. All the variable data (the request file, partner list, etc.) of this instance continue to exist and can be re-activated by repeating the CREATE-FT-INSTANCE command. Any attempt to access a deleted instance is denied with FTR0236.

  The default instance cannot be deleted.

- Setting an instance

  Using the SET-FT-INSTANCE command, you can select the *open*FT instance with which you would like to work (see the user guide). This setting is then valid for all the SDF commands set under this task or program interface calls and remains valid until the task is ended or until the next SET-FT-INSTANCE command. If you want to continue to work with the set instance in a Posix shell then it is necessary to call the following command after starting the shell:

  . ftseti

  The dot (.) followed be a blank is mandatory!

  It is therefore advisable to record this command in the */etc/profile* file.
  If no SET-FT-INSTANCE command is given in a task, then work proceeds using the default instance.

- Displaying instance information

  Using the SHOW-FT-INSTANCE command, you can request information regarding the instances, see the user guide.

- Set or display the BCAM host

  Using the MODIFY-FT-OPTIONS ...,HOST-NAME command, you can assign the current instance a BCAM host. This BCAM host will be used for communication of *open*FT. By doing this, an instance allows itself to be assigned a fixed transport address, which is independent of the computer on which the instance is running.

  On executing the SHOW-FT-OPTIONS command, the name of the BCAM host with which the instance is working is displayed.

- Importing an instance to another computer

  The following steps are required to change over an *open*FT instance to another computer:

  – Stop the instance on the original computer (/STOP-FT).

  – Unload the instance on the original computer (/DELETE-FT-INSTANCE). This unlocks all of the files required by openFT (request file, transfer files, etc.).

  – Import the variable files, the network address (virtual BCAM host) and all of the files required by the requests to the destination computer. This can contain, among other things, the switching over of one or several pubsets).

    It is recommended to import all files of the configuration user ID when changing over.

  – Load the instance on the destination computer (/CREATE-FT-INSTANCE).

  – Start the instance on the destination computer (if this does not occur automatically, then use /SET-FT-INSTANCE, /START-FT).

  After importing an instance to another computer, *open*FT finishes the (under some circumstances restartable) requests, whose admissions were already checked before importing. The new environment must have the same prerequisites as the old computer (the same IDs with the same file access admissions).

  All pubsets that are accessed by requests must be available. All requests whose pubsets are not accessible during restart attempts are aborted.

  On the new computer, the network view must be the same as that on the old computer. This means that, from the point of view of the BCAM, the same host names for partner computers must be available and they must refer to the same partner computer. The network address of the (virtual) host on which the instance is running, must be seen from the outside the same as from the address of the host, on which the instance was previously running.

  The name of the instance must be the same on all of the computers, since, for example, it is used for qualifying temporary files.

3.9 Diagnostics

3.9.1 Controlling the trace function

The FT administrator uses the following commands to control the trace function:

MODIFY-FT-OPTIONS	Modify operating parameters
MODIFY-FT-PARTNER	Modify partner characteristics
SHOW-FT-OPTIONS	Information about operating parameters
SHOW-FT-PARTNERS	Information about partner systems

The FT trace function can be switched on and off irrespective of whether the FT system is active or inactive.

You can set the scope of *open*FT traces globally using the MODIFY-FT-OPTIONS command. You can differentiate by partner type (*open*FT, ftp, FTAM), request type (local/remote and synchronous/asynchronous) and trace scope (with/without file contents). The global setting can be modified on a partner-specific basis using MODIFY-FT-PARTNER.
The following table illustrates four typical cases of trace use.

MODIFY-FT-OPTIONS	**MODIFY-FT-PARTNER**	**Task**	**Effect**
TRACE=*ON	TRACE= *BY-FT-OPTIONS	General monitoring of FT operations.	FT operation is fully monitored.
TRACE=(SWITCH=ON, OPTIONS= NO-BULK-DATA)	TRACE= *BY-FT-OPTIONS	Connect monitoring for all *open*FT partners.	Mass data transfers are not recorded. Recommended for long-lived traces.
TRACE=(SWITCH=ON ,PART-SELECTION= *FTP)	TRACE= *BY-FT-OPTIONS	Monitoring of a a certain type of partner over an extended period. (here, ftp partners)	All events relating to a selected partner type are logged. Despite the extended period, the trace volume does not become excessive.
TRACE=(SWITCH=ON ,REQ-SELECTION= *REM)	TRACE= *BY-FT-OPTIONS	Monitoring of a specific type of request (here, requests submitted by a remote system)	All events relating to certain request types are logged. Despite the extended period, the trace volume does not become excessive.

The default value for MODIFY-FT-PARTNER is BY-FT-OPTIONS. The global settings are thus taken over from MODIFY-FT-OPTIONS.

The following table indicates the interrelations between the most important MODIFY-FT-OPTIONS and MODIFY-FT-PARTNER trace settings.

MODIFY-FT-OPTIONS	**MODIFY-FT-PARTNER**	**Effect**
TRACE=*OFF	equals	*OFF
TRACE=*ON	TRACE=*BY-FT-OPTIONS	*ON
	TRACE=*UNCHANGED	Setting retained
	TRACE=*ON	*ON
	TRACE=*OFF	*OFF
TRACE=(SWITCH=ON, PARTNER-SELECTION= partner type)	TRACE=*BY-FT-OPTIONS	*ON if suitable partner type *OFF if unsuitable partner type
	TRACE=*UNCHANGED	Setting retained
	TRACE=*ON	*ON
	TRACE=*OFF	*OFF
TRACE=(SWITCH=ON, REQUEST-SELECTION= request type)	TRACE=*BY-FT-OPTIONS	*ON if suitable request type *OFF if unsuitable request type
	TRACE=*UNCHANGED	Setting retained
	TRACE=*ON	as *BY-FT-OPTIONS
	TRACE=*OFF	*OFF

3.9.2 Evaluating traces

*open*FT generates trace files for the configuration user ID of the *open*FT instance (default: $SYSFJAM). The file names end with the suffix .FTTF and contain the TSN of the traced task together with the date and time. The first letter of the file name (C/S/Y) specifies whether a control task (C), server task (S) or user task (Y) was traced.
The trace files contain *open*FT, FTAM and FTP requests that have been processed in the corresponding task.

Traces are evaluated with START-FTTRACE:

START-FTTRACE
INPUT = <filename 1..54> ,**OUTPUT** = <filename 1..54> ,**TRACE-OPTION** = <c-string 1..20> ,**SHOW-FILE** = ***YES** / ***NO** ,**PRINT-FILE** = ***NO** / ***YES**

Operand description

INPUT = <filename 1..54>
Filename of the trace file to be evaluated $SYSFJAM.SYSFLF.D*yymmdd*.T*hhmmss.tsn.*

OUTPUT = <filename 1..54>
Filename of the output file.

TRACE-OPTION = <c-string 1..20>
Specifies the options for the trace evaluation.

SHOW-FILE =
Specifies if the evaluated trace file should be displayed with the command SHOW-FILE.

SHOW-FILE = *YES
The evaluated trace file is displayed using the command SHOW-FILE.

SHOW-FILE = *NO
The evaluated trace file is not displayed using the command SHOW-FILE.

PRINT-FILE =
Specifies if the evaluated trace file should be printed.

PRINT-FILE = *NO
The evaluated trace file is not printed.

PRINT-FILE = *YES
The evaluated trace file is printed.

3.9.3 Creating diagnostic records

If, despite due care and attention, an error occurs that neither the FT administrator nor the BS2000 system administrator can rectify, contact your Service Center. To facilitate trouble-shooting, please submit the following:

– detailed description of the error situation and statement indicating whether the error is reproducible;

– trace files;

– if applicable the result list of the request that triggered the error;

– complete tracer log of the entire session or the SYS.CONSLOG file (also from partner system is possible);

– general information as for BS2000 system error on *open*FT or BS2000/OSD, DCAM, PDN, PLAM, SDF and, if required, *open*FT-FTAM, *open*FT-AC for BS2000, NFS and POSIX:

  1. system version number,
  2. loader - subversion number / code,
  3. list of all rep corrections used;

– version of the FT partner and details of the transport system (e.g. DCAM, CCP / CMX, VTAM, etc.);

– system dumps requested under the TSN FJAM, FTAM, NDMS or FT server tasks;

– system dumps after interrupts in the modules of the FT and FTAC subsystems).

The SHOW-FT-DIAG command can be used to output any diagnostic codes written when the error occurred (together with time and date). In this case, SHOW-FT-DIAG supplies the following output:

```
/sh-ft-diag
% DATE           TIME      SSID  COMPONENT    FUNCTION        INFO
% 20061021       143307    FT    79/yfasdia   3/EuisyMsg      fd00000c
```

4 Administration commands

*open*FT for BS2000 offers a new SDF command interface for administration.
The FT administration commands can be issued from the console. Administration from the terminal requires the FT-ADMINISTRATION privilege, which is assigned by default to the TSOS ID. If SECOS is in use, this privilege can also be assigned to other user IDs. See the SECOS manual for details.
The FT administrator commands that may be entered via the console, can also be set by all the IDs with the OPERATING privilege. If necessary, this privilege can be taken away from these IDs.
These are the commands: ADD-FT-PARTNER, CREATE-FT-INSTANCE, CREATE-FT-KEY-SET, DELETE-FT-INSTANCE, DELETE-FT-KEY-SET, MODIFY-FT-INSTANCE, MODIFY-FT-OPTIONS, MODIFY-FT-PARTNER, REMOVE-FT-PARTNER, SHOW-FT-OPTIONS, SHOW-FT-PARTNER, START-FT, STOP-FT, UPDATE-FT-PUBLIC-KEYS.

4.1 Functional command overview

The following section provides an overview of the FT and FTAC commands as they relate to each function.

4.1.1 FT commands

These administration commands can be used to:

Enter, modify, delete partners in/from the partner list

ADD-FT-PARTNER page 105
REMOVE-FT-PARTNER page 223
MODIFY-FT-PARTNER page 194

Activate *open*FT

START-FT page 288

Deactivate *open*FT

STOP-FT page 291

Monitor *open*FT and control the resources used

MODIFY-FT-OPTIONS page 178

Obtain information on *open*FT

SHOW-FT-OPTIONS page 263
SHOW-FT-PARTNERS page 271

Obtain information on FT requests

SHOW-FILE-TRANSFER page 224

Cancel FT requests

CANCEL-FILE-TRANSFER page 117

Modify FT request queue

MODIFY-FILE-TRANSFER page 164

Show and delete FT and FTAC logging records

SHOW-FT-LOGGING-RECORDS page 249
DELETE-FT-LOGGING-RECORDS page 152

Activate/deactivate requests with certain partner systems, modify addresses or names

MODIFY-FT-PARTNER page 194

Administer key pair sets for authentication

CREATE-FT-KEY-SET page 125
UPDATE-FT-PUBLIC-KEYS page 292
DELETE-FT-KEY-SET page 150

Administer *open*FT instances

CREATE-FT-INSTANCE page 123
DELETE-FT-INSTANCE page 149
MODIFY-FT-INSTANCE page 176

4.1.2 FTAC commands

*open*FT-AC must be installed in order to use the following commands.

Edit FTAC admission profiles

CREATE-FT-PROFILE	page 127
DELETE-FT-PROFILE	page 156
MODIFY-FT-PROFILE	page 199
SHOW-FT-PROFILE	page 279

Edit FTAC admission sets

MODIFY-FT-ADMISSION-SET	page 169
SHOW-FT-ADMISSION-SET	page 244

Store and display saved FTAC admission profiles and sets

EXPORT-FTAC-ENVIRONMENT	page 159
IMPORT-FTAC-ENVIRONMENT	page 161
SHOW-FTAC-ENVIRONMENT	page 240

Display partner systems

SHOW-FT-RANGE	page 285

4.2 Entering FT commands

Please remember the following when entering commands:

– You must insert commas to separate the individual operands of a command, e.g.

```
/TRANSFER-FILE TRANSFER-DIRECTION=TO,PARTNER=CENTER,LOCAL-PARAMETER=...
```

– Values assigned within quotes must be quoted again, i.e., the number of quotes is duplicated.

– If there is no default value marked (by underscoring) for an operand, then it **must** be specified with a valid value (mandatory operand).

– The command and operand names consist of keywords which are separated by hyphens. Constant operand values can also be considered to be keywords while variable operand values are data types.

– A distinction is made between positional operands and keyword operands. Positional operands are uniquely determined by their position in the command while keyword operands are uniquely determined by their keyword, for example `TRANSFER-DIRECTION=...` There are a number of considerations to be borne in mind when specifying such operands (see below).

– You can abbreviate your entries for commands and operands, always ensuring that your entries retain their unequivocality. You can also use positional operands if you wish. Short forms and long forms can be mixed at will. Certain abbreviated forms of keywords and a number of positional operands are guaranteed for *open*FT. This means that you will find these options unchanged in subsequent versions. This means, therefore, that to be “on the safe side”, you should form the habit of entering these commands in their abbreviated form. You should take particular care to use the guaranteed abbreviated forms in procedures, as this will ensure their continued executability in subsequent versions. The recommended abbreviations are used in the examples shown in this chapter, and the possible abbreviations are listed for the individual command formats.

– If a structure is preceded by an introductory operand value (e.g. *BS2000 is an introductory operand value in “REM=*BS2000(...)”), then the opening parentheses must immediately follow this operand. Introductory operand value may be omitted if there is no risk of ambiguity.

– The asterisk (*) that precedes constant operand values may be omitted if there is no risk of ambiguity. Please ensure that it is not a guaranteed abbreviation.

When you enter commands, the value assignments for the operands may be specified in positional form, in keyword form or in mixed form.
Please note the following:

– When you perform value assignments in positional form, the first value is assigned to the first operand in the command, the second value to the second operand etc.

– Values assigned in positional form are separated by commas. You must also enter a comma for each operand for which no value is assigned.

– If two values are assigned to an operand, the last value to be assigned always applies. This also applies to parameter specifications in introductory operand values within the corresponding structure brackets. However, for the sake of clarity, double assignments should generally be avoided.

– If you mix the different forms of operand value assignments (positional and keyword form), then you must observe the correct sequence. Note that you can start your input with positional operands and follow these with keyword operands but not the other way round!

– Since there is a possibility that the sequence of operands may change in subsequent versions, only keyword operands should be used in procedures.

4.3 SDF syntax representation

The following example shows the representation of the syntax of a command in a manual. The command format consists of a field with the command name. All operands with their legal values are then listed. Operand values which introduce structures and the operands dependent on these operands are listed separately.

HELP-SDF Alias: **HPSDF**

GUIDANCE-**MODE** = ***NO** / ***Y**ES
,**SDF-COM**MANDS = ***NO** / ***Y**ES
,**ABBR**EVIATION-**RULES** = ***NO** / ***Y**ES
,**GUID**ED-**DIA**LOG = ***Y**ES(...)
***Y**ES(...)
SCREEN-STEPS = ***NO** / ***Y**ES
,**SPEC**IAL-**FUNC**TIONS = ***NO** / ***Y**ES
,**FUNC**TION-**KEYS** = ***NO** / ***Y**ES
,**NEXT-FIELD** = ***NO** / ***Y**ES
,**UNGUID**ED-**DIA**LOG = ***Y**ES(...) / ***NO**
***Y**ES(...)
SPECIAL-**FUNC**TIONS = ***NO** / ***Y**ES
,**FUNC**TION-**KEYS** = ***NO** / ***Y**ES

Representation of the syntax of the user command HELP-SDF

This syntax description is valid for SDF V4.6A.The syntax of the SDF command/statement language is explained in the following three tables.

table 1: Notational conventions

The meanings of the special characters and the notation used to describe command and statement formats are explained in Table 1.

table 2: Data types

Variable operand values are represented in SDF by data types. Each data type represents a specific set of values. The number of data types is limited to those described in Table 2.

The description of the data types is valid for the entire set of commands/statements. Therefore only deviations (if any) from the attributes described here are explained in the relevant operand descriptions.

table 3: Suffixes for data types

Data type suffixes define additional rules for data type input. They contain a length or interval specification and can be used to limit the set of values (suffix begins with *without*), extend it (suffix begins with *with*), or declare a particular task mandatory (suffix begins with *mandatory*). The following short forms are used in this manual for data type suffixes:

cat-id	cat
completion	compl
correction-state	corr
generation	gen
lower-case	low
manual-release	man
odd-possible	odd
path-completion	path-compl
separators	sep
temporary-file	temp-file
under-score	under
user-id	user
version	vers
wildcard-constr	wild-constr
wildcards	wild

The description of the ‘integer’ data type in Table 3 contains a number of items in italics; the italics are not part of the syntax and are only used to make the table easier to read.
For special data types that are checked by the implementation, Table 3 contains suffixes printed in italics (see the *special* suffix) which are not part of the syntax.

The description of the data type suffixes is valid for the entire set of commands/statements. Therefore only deviations (if any) from the attributes described here are explained in the relevant operand descriptions.

Metasyntax

Representation	Meaning	Examples
UPPERCASE LETTERS	Uppercase letters denote keywords (command, statement or operand names, keyword values) and constant operand values. Keyword values begin with *	**HELP-SDF** **SCREEN-STEPS** = ***NO**
UPPERCASE LETTERS in boldface	Uppercase letters printed in boldface denote guaranteed or suggested abbreviations of keywords.	**GUID**ANCE-**MODE** = ***Y**ES
=	The equals sign connects an operand name with the associated operand values.	**GUID**ANCE-**MODE** = ***NO**
< >	Angle brackets denote variables whose range of values is described by data types and suffixes (see Tables 2 and 3).	**SYNTAX-F**ILE = <filename 1..54>
Underscoring	Underscoring denotes the default value of an operand.	**GUID**ANCE-**MODE** = ***NO**
/	A slash serves to separate alternative operand values.	**NEXT-FIELD** = ***NO** / ***Y**ES
(…)	Parentheses denote operand values that initiate a structure.	,**UNGUID**ED-**DIA**LOG = ***Y**ES(...) / ***NO**
[ ]	Square brackets denote operand values which introduce a structure and are optional. The subsequent structure can be specified without the initiating operand value.	**SELECT** = [***BY-ATTR**IBUTES](...)
Indentation	Indentation indicates that the operand is dependent on a higher-ranking operand.	,**GUID**ED-**DIA**LOG = ***Y**ES(...) ***Y**ES(...) **SCREEN-STEPS** = ***NO** / ***Y**ES

Table 1: Metasyntax (part 1 of 2)

Representation	Meaning	Examples
|	A vertical bar identifies related operands within a structure. Its length marks the beginning and end of a structure. A structure may contain further structures. The number of vertical bars preceding an operand corresponds to the depth of the structure.	**SUP**PORT = ***TAPE**(...) ***TAPE(...)** | **VOL**UME = ***ANY**(...) | ***ANY**(...) | | ...
,	A comma precedes further operands at the same structure level.	**GUID**ANCE-**MODE** = ***NO** / ***Y**ES ,**SDF-COM**MANDS = ***NO** / ***Y**ES
list-poss(n):	The entry “list-poss” signifies that a list of operand values can be given at this point. If (n) is present, it means that the list must not have more than n elements. A list of more than one element must be enclosed in parentheses.	list-poss: ***SAM** / ***ISAM** list-poss(40): <structured-name 1..30> list-poss(256): ***OMF** / ***SYSLST**(...) / <filename 1..54>
Alias:	The name that follows represents a guaranteed alias (abbreviation) for the command or statement name.	**HELP-SDF** Alias: **HPSDF**

Table 1: Metasyntax (part 2 of 2)

Data types

Data type	Character set	Special rules
alphanum-name	A…Z 0…9 $, #, @	
cat-id	A…Z 0…9	Not more than 4 characters; must not begin with the string PUB
command-rest	freely selectable	
composed-name	A…Z 0…9 $, #, @ hyphen period catalog ID	Alphanumeric string that can be split into multiple substrings by means of a period or hyphen. If a file name can also be specified, the string may begin with a catalog ID in the form :cat: (see data type filename).
c-string	EBCDIC character	Must be enclosed within single quotes; the letter C may be prefixed; any single quotes occurring within the string must be entered twice.
date	0…9 Structure identifier: hyphen	Input format: yyyy-mm-dd yyyy: year; optionally 2 or 4 digits mm:month dd:day
device	A…Z 0…9 hyphen	Character string, max. 8 characters in length, corresponding to a device available in the system. In guided dialog, SDF displays the valid operand values. For notes on possible devices, see the relevant operand description.
fixed	+, - 0…9 period	Input format: [sign][digits].[digits] [sign]:+ or - [digits]:0...9 must contain at least one digit, but may contain up to 10 characters (0...9, period) apart from the sign.

Table 2: Data types (part 1 of 6)

Data type	Character set	Special rules
filename	A…Z 0…9 $, #, @ hyphen period	Input format: [:cat:][$user.] { file | file(no) | group | group { (*abs) | (+rel) | (-rel) } } :cat: optional entry of the catalog identifier; character set limited to A...Z and 0...9; maximum of 4 characters; must be enclosed in colons; default value is the catalog identifier assigned to the user ID, as specified in the user catalog. $user. optional entry of the user ID; character set is A…Z, 0…9, $, #, @; maximum of 8 characters; first character cannot be a digit; $ and period are mandatory; default value is the user's own ID. $. (special case) system default ID file file or job variable name; may be split into a number of partial names using a period as a delimiter: $name_1$[.$name_2$[...]] $name_i$ does not contain a period and must not begin or end with a hyphen; file can have a maximum length of 41 characters; it must not begin with a $ and must include at least one character from the range A...Z.

Table 2: Data types (part 2 of 6)

Data type	Character set	Special rules

filename
(continued)
		#file (special case)
		@file (special case)
		# or @ used as the first character indicates temporary files or job variables, depending on system generation.
		file(no)
		tape file name
		no: version number;
		character set is A...Z, 0...9, $, #, @.
		Parentheses must be specified.
		group
		name of a file generation group
		(character set: as for “file”)
		group { (*abs) (+rel) (-rel) }
		(*abs)
		absolute generation number (1-9999);
		* and parentheses must be specified.
		(+rel)
		(-rel)
		relative generation number (0-99);
		sign and parentheses must be specified.

integer	0…9, +, -	+ or -, if specified, must be the first character.
name	A…Z 0…9 $, #, @	Must not begin with 0...9.

Table 2: Data types (part 3 of 6)

Data type	Character set	Special rules
partial-filename	A…Z 0…9 $, #, @ hyphen period	Input format: [:cat:][$user.][partname.] :cat:see filename $user.see filename partname optional entry of the initial part of a name common to a number of files or file generation groups in the form: $name_1$.[$name_2$.[...]] $name_i$ (see filename). The final character of “partname” must be a period. At least one of the parts :cat:, $user. or partname must be specified.
posix-filename	A...Z 0...9 special characters	String with a length of up to 255 characters if entered as a C string, or 219 characters if entered in *POSIX syntax. It consists of either one or two periods or of alphanumeric characters and special characters. The special characters must be escaped with a preceding \ (backslash); the / is not allowed. Must be enclosed within single quotes if alternative data types are permitted, separators are used, or the first character is a ?, ! or ^ A distinction is made between uppercase and lowercase.
posix-pathname	A...Z 0...9 special characters structure identifier: slash	Input format: [/]$part_1$/.../$part_n$ where $part_i$ is a posix-filename; max. 1023 characters in C string syntax and 219 in *POSIX syntax; must be enclosed within single quotes if alternative data types are permitted, separators are used, or the first character is a ?, ! or ^

Table 2: Data types (part 4 of 6)

Data type	Character set	Special rules
product-version	A…Z 0…9 period single quote	Input format: [[C]'][V][m]m.naso['] correction status release status where m, n, s and o are all digits and a is a letter. Whether the release and/or correction status may/must be specified depends on the suffixes to the data type (see the suffixes without-corr, without-man, mandatory-man and mandatory-corr in Table 3). product-version may be enclosed within single quotes (possibly with a preceding C). The specification of the version may begin with the letter V.
structured-name	A…Z 0…9 $, #, @ hyphen	Alphanumeric string which may comprise a number of substrings separated by a hyphen. First character: A...Z or $, #, @
text	freely selectable	For the input format, see the relevant operand descriptions.
time	0…9 structure identifier: colon	Time-of-day entry: Input format: { hh:mm:ss / hh:mm / hh } { hh:hours / mm:minutes / ss:seconds } Leading zeros may be omitted

Table 2: Data types (part 5 of 6)

Data type	Character set	Special rules
vsn	a) A…Z 0…9	a) Input format: pvsid.sequence-no max. 6 characters pvsid: 2-4 characters; PUB must not be entered sequence-no: 1-3 characters
	b) A…Z 0…9 $, #, @	b) Max. 6 characters; PUB may be prefixed, but must not be followed by $, #, @.
x-string	Hexadecimal: 00…FF	Must be enclosed in single quotes; must be prefixed by the letter X. There may be an odd number of characters.
x-text	Hexadecimal: 00…FF	Must not be enclosed in single quotes; the letter X must not be prefixed. There may be an odd number of characters.

Table 2: Data types (part 6 of 6)

Suffixes for data types

Suffix	Meaning
x..y *unit*	With data type “integer”: interval specification x minimum value permitted for “integer”. x is an (optionally signed) integer. y maximum value permitted for “integer”. y is an (optionally signed) integer. *unit* with “integer” only: additional units. The following units may be specified: *days* *byte* *hours* *2Kbyte* *minutes* *4Kbyte* *seconds* *Mbyte* milliseconds
x..y *special*	With the other data types: length specification For data types catid, date, device, product-version, time and vsn the length specification is not displayed. x minimum length for the operand value; x is an integer. y maximum length for the operand value; y is an integer. x=y the length of the operand value must be precisely x. *special* Specification of a suffix for describing a special data type that is checked by the implementation. “special” can be preceded by other suffixes. The following specifications are used: *arithm-expr* arithmetic expression (SDF-P) *bool-expr* logical expression (SDF-P) *string-expr* string expression (SDF-P) *expr* freely selectable expression (SDF-P) *cond-expr* conditional expression (JV) *symbol* CSECT or entry name (BLS)
with	Extends the specification options for a data type.
-compl	When specifying the data type “date”, SDF expands two-digit year specifications in the form yy-mm-dd to: 20jj-mm-tt if jj < 60 19jj-mm-tt if jj ≥ 60
-low	Uppercase and lowercase letters are differentiated.
-path-compl	For specifications for the data type “filename”, SDF adds the catalog and/or user ID if these have not been specified.

Table 3: Data type suffixes (part 1 of 7)

<table>
<tr><th>Suffix</th><th>Meaning</th></tr>
<tr><td>with (contd.)</td><td></td></tr>
<tr><td>-under</td><td>Permits underscores (_) for the data types “name” and “composed-name”.</td></tr>
<tr><td>-wild(n)</td><td>Parts of names may be replaced by the following wildcards.
n denotes the maximum input length when using wildcards.
Due to the introduction of the data types posix-filename and posix-pathname, SDF now accepts wildcards from the UNIX world (referred to below as POSIX wildcards) in addition to the usual BS2000 wildcards. However, as not all commands support POSIX wildcards, their use for data types other than posix-filename and posix-pathname can lead to semantic errors.
Only POSIX wildcards or only BS2000 wildcards should be used within a search pattern. Only POSIX wildcards are allowed for the data types posix-filename and posix-pathname. If a pattern can be matched more than once in a string, the first match is used.
<table>
<tr><th>BS2000 wildcards</th><th>Meaning</th></tr>
<tr><td>*</td><td>Replaces an arbitrary (even empty) character string. If the string concerned starts with *, then the * must be entered twice in succession if it is followed by other characters and if the character string entered does not contain at least one other wildcard.</td></tr>
<tr><td>Terminating period</td><td>Partially-qualified entry of a name.
Corresponds implicitly to the string “./*”, i.e. at least one other character follows the period.</td></tr>
<tr><td>/</td><td>Replaces any single character.</td></tr>
<tr><td><s_x:s_y></td><td>Replaces a string that meets the following conditions:
– It is at least as long as the shortest string (s_x or s_y)
– It is not longer than the longest string (s_x or s_y)
– It lies between s_x and s_y in the alphabetic collating sequence; numbers are sorted after letters (A...Z0...9)
– s_x can also be an empty string (which is in the first position in the alphabetic collating sequence)
– s_y can also be an empty string, which in this position stands for the string with the highest possible code (contains only the characters X'FF')</td></tr>
<tr><td><s_1,…></td><td>Replaces all strings that match any of the character combinations specified by s. s may also be an empty string. Any such string may also be a range specification “s_x:s_y” (see above).</td></tr>
</table>
</td></tr>
</table>

Table 3: Data type suffixes (part 2 of 7)

Suffix	Meaning	
with-wild(n)		
(continued)	-s	Replaces all strings that do not match the specified string s. The minus sign may only appear at the beginning of string s. Within the data types filename or partial-filename the negated string -s can be used exactly once, i.e. -s can replace one of the three name components: cat, user or file.
	Wildcards are not permitted in generation and version specifications for file names. Only system administration may use wildcards in user IDs. Wildcards cannot be used to replace the delimiters in name components cat (colon) and user ($ and period).	
	POSIX wildcards	Meaning
	*	Replaces any single string (including an empty string). An * appearing at the first position must be duplicated if it is followed by other characters and if the entered string does not include at least one further wildcard.
	?	Replaces any single character; not permitted as the first character outside single quotes.
	$[c_x\text{-}c_y]$	Replaces any single character from the range defined by c_x and c_y, including the limits of the range. c_x and c_y must be normal characters.
	[s]	Replaces exactly one character from string s. The expressions $[c_x\text{-}c_y]$ and [s] can be combined into $[s_1c_x\text{-}c_ys_2]$.
	$[!c_x\text{-}c_y]$	Replaces exactly one character not in the range defined by c_x and c_y, including the limits of the range. c_x and c_y must be normal characters. The expressions $[!c_x\text{-}c_y]$ and [!s] can be combined into $[!s_1c_x\text{-}c_ys_2]$.
	[!s]	Replaces exactly one character not contained in string s. The expressions [!s] and $[!c_x\text{-}c_y]$ can be combined into $[!s_1c_x\text{-}c_ys_2]$.

Table 3: Data type suffixes (part 3 of 7)

<table>
<tr><th>Suffix</th><th>Meaning</th></tr>
<tr><td>with (contd.)

-wild-
constr(n)</td><td>Specification of a constructor (string) that defines how new names are to be constructed from a previously specified selector (i.e. a selection string with wildcards). See also with-wild. n denotes the maximum input length when using wildcards.
The constructor may consist of constant strings and patterns. A pattern (character) is replaced by the string that was selected by the corresponding pattern in the selector.
The following wildcards may be used in constructors:
<table>
<tr><td>Wildcard</td><td>Meaning</td></tr>
<tr><td>*</td><td>Corresponds to the string selected by the wildcard * in the selector.</td></tr>
<tr><td>Termina-
ting period</td><td>Corresponds to the partially-qualified specification of a name in the selector;
corresponds to the string selected by the terminating period in the selector.</td></tr>
<tr><td>/ or ?</td><td>Corresponds to the character selected by the / or ? wildcard in the selector.</td></tr>
<tr><td><n></td><td>Corresponds to the string selected by the n-th wildcard in the selector, where n is an integer.</td></tr>
</table>
Allocation of wildcards to corresponding wildcards in the selector:
All wildcards in the selector are numbered from left to right in ascending order (global index).
Identical wildcards in the selector are additionally numbered from left to right in ascending order (wildcard-specific index).
Wildcards can be specified in the constructor by one of two mutually exclusive methods:
1. Wildcards can be specified via the global index: <n>
2. The same wildcard may be specified as in the selector; substitution occurs on the basis of the wildcard-specific index. For example:
the second “/” corresponds to the string selected by the second “/” in the selector</td></tr>
</table>

Table 3: Data type suffixes (part 4 of 7)

Suffix	Meaning
with-wild-constr (continued)	The following rules must be observed when specifying a constructor: – The constructor can only contain wildcards of the selector. – If the string selected by the wildcard <...> or [...] is to be used in the constructor, the index notation must be selected. – The index notation must be selected if the string identified by a wildcard in the selector is to be used more than once in the constructor. For example: if the selector “A/” is specified, the constructor “A<n><n>” must be specified instead of “A//”. – The wildcard * can also be an empty string. Note that if multiple asterisks appear in sequence (even with further wildcards), only the last asterisk can be a non-empty string, e.g. for “****” or “*//*”. – Valid names must be produced by the constructor. This must be taken into account when specifying both the constructor and the selector. – Depending on the constructor, identical names may be constructed from different names selected by the selector. For example: “A/*” selects the names “A1” and “A2”; the constructor “B*” generates the same new name “B” in both cases. To prevent this from occurring, all wildcards of the selector should be used at least once in the constructor. – If the selector ends with a period, the constructor must also end with a period (and vice versa).

Table 3: Data type suffixes (part 5 of 7)

<table>
<tr><th>Suffix</th><th>Meaning</th></tr>
<tr><td>with-wild-constr (continued)</td><td>Examples:
<table>
<tr><th>Selector</th><th>Selection</th><th>Constructor</th><th>New name</th></tr>
<tr><td>A//*</td><td>AB1
AB2
A.B.C</td><td>D<3><2></td><td>D1
D2
D.CB</td></tr>
<tr><td>C.<A:C>/<D,F></td><td>C.AAD
C.ABD
C.BAF
C.BBF</td><td>G.<1>.<3>.XY<2></td><td>G.A.D.XYA
G.A.D.XYB
G.B.F.XYA
G.B.F.XYB</td></tr>
<tr><td>C.<A:C>/<D,F></td><td>C.AAD
C.ABD
C.BAF
C.BBF</td><td>G.<1>.<2>.XY<2></td><td>G.A.A.XYA
G.A.B.XYB
G.B.A.XYA
G.B.B.XYB</td></tr>
<tr><td>A//B</td><td>ACDB
ACEB
AC.B
A.CB</td><td>G/XY/</td><td>GCXYD
GCXYE
GCXY. [1]
G.XYC</td></tr>
</table>
[1] The period at the end of the name may violate naming conventions (e.g. for fully-qualified file names).</td></tr>
<tr><td>without</td><td>Restricts the specification options for a data type.</td></tr>
<tr><td>-cat</td><td>Specification of a catalog ID is not permitted.</td></tr>
<tr><td>-corr</td><td>Input format: [[C]'][V][m]m.na[']
Specifications for the data type product-version must not include the correction status.</td></tr>
<tr><td>-gen</td><td>Specification of a file generation or file generation group is not permitted.</td></tr>
<tr><td>-man</td><td>Input format: [[C]'][V][m]m.n[']
Specifications for the data type product-version must not include either release or correction status.</td></tr>
<tr><td>-odd</td><td>The data type x-text permits only an even number of characters.</td></tr>
<tr><td>-sep</td><td>With the data type “text”, specification of the following separators is not permitted: ; = ( ) < > ␣ (i.e. semicolon, equals sign, left and right parentheses, greater than, less than, and blank).</td></tr>
<tr><td>-temp-file</td><td>Specification of a temporary file is not permitted (see #file or @file under filename).</td></tr>
</table>

Table 3: Data type suffixes (part 6 of 7)

Suffix	Meaning
without (cont.)	
-user	Specification of a user ID is not permitted.
-vers	Specification of the version (see “file(no)”) is not permitted for tape files.
-wild	The file types posix-filename and posix-pathname must not contain a pattern (character).
mandatory	Certain specifications are necessary for a data type.
-corr	Input format: [[C]'][V][m]m.naso['] Specifications for the data type product-version must include the correction status and therefore also the release status.
-man	Input format: [[C]'][V][m]m.na[so]['] Specifications for the data type product-version must include the release status. Specification of the correction status is optional if this is not prohibited by the use of the suffix without-corr.
-quotes	Specifications for the data types posix-filename and posix-pathname must be enclosed in single quotes.

Table 3: Data type suffixes (part 7 of 7)

How are the operands described?

After the format of each command there is a detailed description of all the operands, the possible value assignments and their functions.

Otherwise the same metasyntax is used in describing operands as in the representation of the command formats (see above).

4.4 Command return codes

The *open*FT commands supply return codes that you can query when using SDF-P. Each return code consists of a subcode1 (SC1), a subcode2 (SC2) and the maincode (MC).

Subcode1

Subcode1 represents the error class and is a decimal number. The possible error classes are:

– No error:
  the value of subcode1 is 0.
– Syntax error:
  the value of subcode1 is between 1 and 31, inclusive.
– Internal error (system error):
  the value of subcode1 is 32.
– Errors not assigned to any other class:
  the value of subcode1 is between 64 and 127, inclusive. If the value of subcode 1 is in this range, the maincode must be evaluated in order to ascertain the appropriate action.
– Command cannot be executed at this time:
  the value of subcode1 is between 128 and 130, inclusive.

Subcode2

Subcode2 either contains information supplementary to that in subcode1 or is equal to 0.

Maincode

The maincode corresponds to the message key of the SYSOUT message. You can use the /HELP-MSG-INFORMATION command to fetch detailed information.

The command return codes are always located after the detailed description of the command. In each case, the corresponding section specifies which command return codes are possible and what their meaning is.

You will find the corresponding specifications in the “Administration commands” section of this manual and in the “User commands” section of the *open*FT User Guide.

4.5 OPS variables

With OPS (Output Presentation Service), you have the option to create the outputs of SHOW commands alternative or additional to the output in SYSLST/SYSOUT in OPS variables. For this to be possible, SDF-P must be installed. The user must generate the corresponding OPS variables with DECLARE-VARIABLE. The information supplied by SHOW commands is stored by *open*FT in an SDF-P structure, which can be evaluated with the help of an SDF-P procedure. Structure elements which have not been set due to a corresponding command input are output without value assignment.

The request to set OPS variables is made by integrating the unchanged FT command into the BS2000 command EXEC-CMD.

Example

```
/EXEC-CMD (SHOW-FILE-TRANSFER),TEXT=*N,STRUCT-OUT=<variable-name>
```

The following *open*FT user commands offer OPS support:

– SHOW-FILE-TRANSFER
– SHOW-REMOTE-FILE-ATTRIBUTES
– SHOW-FILE-FT-ATTRIBUTES
– SHOW-FTAC-ENVIRONMENT
– SHOW-FT-ADMISSION-SET
– SHOW-FT-LOGGING-RECORDS
– SHOW-FT-OPTIONS
– SHOW-FT-PARTNERS
– SHOW-FT-PROFILE
– SHOW-FT-RANGE

4.6 Output in CSV format

The output of some SHOW commands in *open*FT and *open*FT-AC can be optionally requested in CSV (Comma Separated Values) format. CSV is a popular format in the PC environment in which tabular data is defined by lines. Output in CSV format is offered for the following commands:

– SHOW-FILE-TRANSFER

– SHOW-REMOTE-FILE-ATTRIBUTES

– SHOW-FILE-FT-ATTRIBUTES

– SHOW-FTAC-ENVIRONMENT

– SHOW-FT-ADMISSION-SET

– SHOW-FT-LOGGING-RECORDS

– SHOW-FT-OPTIONS

– SHOW-FT-PARTNERS

– SHOW-FT-PROFILE

– SHOW-FT-RANGE

Many programs such as spreadsheets, databases, etc., can import data in CSV format. This means that you can use the processing and presentation features of such programs on the data output by the above commands.

The output fields are described in the appendix starting on page 293.

The first line is the header and contains the field names of the respective columns. **Only the field names are guaranteed, not the order of fields in a record.** In other words, the order of columns is determined by the order of the field names in the header line.

One example of a possible evaluation procedure is supplied as a reference template in the Microsoft Excel format under the name $SYSFJAM.FTACCNT.XLT. You will need to first make a binary copy of this template on your PC. The template evaluates a CSV logging file by means of an automatically running macro. The result shows the number of inbound and outbound requests and the Kilobytes transferred in each case for all BS2000 users.

4.7 ADD-FT-PARTNER (FTADDPTN) Add remote system to the partner list

You can add a remote system to the partner list of the local system using the command ADD-FT-PARTNER (or FTADDPTN). The network or the transport system should be generated beforehand. Please refer to the appropriate manuals on PDN and BCAM for further information on the generation process. A transport system in accordance with ISO or TCP/IP can be used for generation.

If dynamic partners are permitted then inbound and outbound requests can be processed with partners which are accessed via their addresses and are not defined in the partner list.

It is also possible to link *open*FT to an SNA network; you will not require a gateway with BCAM V11 or later. Again, you should refer to the appropriate manuals (TRANSIT-SNA, TRANSIT-CD).

You can issue the ADD-FT-PARTNER command for all partner types while the FT system is running (*open*FT partners, FTAM partners and ftp partners).

ADD-FT-PARTNER / **FTADDPTN**

PARTNER-**NAME** = <name 1..8> / ***NONE**

,**PART**NER-**ADDR**ESS = <text 1..200 with-low>

,**SEC**URITY-**LEV**EL = ***STD** / ***BY-PART**NER-**ATTR**IBUTES / <integer 1..100>

,**STATE** = ***ACTIVE**(...) / ***DEACT**

***ACTIVE**(...)
AUTOMATIC-DEACT = ***NO** / ***YES**

,**ID**ENTIFICATION = ***STD** / <composed-name 1..64> / <c-string 1..64 with-low>

,**SESS**ION-**ROUT**ING-**INFO** = ***NONE** / ***ID**ENTIFICATION / <alphanum-name 1..8>

,**PART**NER-**CHECK** = ***BY-FT-OPT**IONS / ***STD** / ***TRANS**PORT-**ADDR**ESS

,**TRACE** = ***BY-FT-OPT**IONS / ***ON** / ***OFF**

,**AUTH-MANDATORY**= ***NO** / ***YES**

Operands

PARTNER-NAME =
Is the symbolic name of the partner system. It can be freely selected and need only be unique within *open*FT.

PARTNER-NAME = <name 1..8>
The operand value “name” consists of a maximum of 8 alphanumeric characters and must be unique in the local system. The FT administrator defines this name. This name can be used in the PARTNER parameter in all FT commands in order to address the partner system.

PARTNER-NAME = *NONE
Specifies that the partner is a dynamic partner.

PARTNER-ADDRESS = <text 1..200 with-low>
Is the address of the partner system. This specifies whether the partner is an *open*FT or FTAM or FTP partner. For more information on address specifications, see section “Specifying partner addresses” on page 39.

SECURITY-LEVEL =
This operand assigns a security level to a remote system.

SECURITY-LEVEL = *STD
If you set this operand to *STD or if you do not enter a value here, a standard security level is assigned to the remote system. This standard security level is defined using the command MODIFY-FT-OPTIONS. You can define a fixed value or specify that the value should be attribute-dependent.

SECURITY-LEVEL = *BY-PARTNER-ATTRIBUTES
If you set the operand to *BY-PARTNER-ATTRIBUTES then the security level is defined automatically. This setting assigns partners that are authenticated by *open*FT the security level 10. Partners that are known in BCAM (i.e. they are addressed via their BCAM names) are assigned the security level 90. All other partners are assigned security level 100.

SECURITY-LEVEL = <integer 1..100>
Must be specified if you wish to assign an individual security level to a specific remote system.

STATE =
You can use this operand to control locally distributed FT requests which are directed to the remote system you have just entered.

STATE = *ACTIVE(...)
Locally distributed FT requests to this remote system are processed.

AUTOMATIC-DEACT =
Defines whether cyclical attempts to establish a connection to a specific partner system are prohibited after a number of attempts by deactivating the partner system.

AUTOMATIC-DEACT = *NO
Failed attempts to establish a connection of this partner system do not result in its deactivation.

AUTOMATIC-DEACT = *YES
Failed attempts to establish a connection of this partner system result in its deactivation. If file transfer is to be resumed with this partner system, it must be explicitly reactivated.

STATE = *DEACT
Locally distributed FT requests to this remote system are not processed (not started) for the moment.

IDENTIFICATION =
Identification of the *open*FT instance in the partner system.

IDENTIFICATION = *STD
The partner address is used as the identification.

IDENTIFICATION = <composed-name 1..64> / <c-string 1..64 with-low>
The network-wide, unique instance ID of the *open*FT instance in the partner system. It is set by the FT administrator of the partner system (for example in BS2000, by using MODIFY-FT-OPTIONS IDENTIFICATION=, in UNIX systems or Windows systems, by using ftmodo -id). The uniqueness of this ID must be based on something other than case-sensitivity. An instance ID may be comprised of alphanumeric characters or special characters. It is advisable only to use the special characters ".", "-", ":" or "%". The initial character must be alphanumeric or the special character "%". The "%" character may only be used as an initial character. An alphanumeric character must follow the "." character.

For more details on allocating instance IDs, please refer to page 46.

You should always specify the instance identification of the partner system explicitly and should not use the default value (IDENTIFICATION=*STD). The explicit specification improves the performance.

SESSION-ROUTING-INFO =
If the partner system is only accessible by a go-between instance (for example *open*FTIF gateway), specify the address information that the gateway instance uses for re-routing here. This is necessary, for example, for partner systems using *open*FT for OS/390 and z/OS, dependent on TRANSIT coupling.

SESSION-ROUTING-INFO = *NONE
By default, no specification is required.
The session selector can be specified as a part of the partner address.

SESSION-ROUTING-INFO = *IDENTIFICATION
Connections to the partner are re-routed via a gateway that supports the instance ID as address information.

SESSION-ROUTING-INFO = <alphanum-name 1..8>
Connections to the partner are re-routed via a gateway that supports the specified character string as address information.

PARTNER-CHECK =
Use this parameter to modify the global settings for the sender check in a partner-specific way. These settings are only valid for *open*FT partners that do not work with authentication (for example, partners with *open*FT V8.0 or earlier). For FTAM partners, the sender check is enabled.

PARTNER-CHECK = *BY-FT-OPTIONS
The global settings are valid for the partners.

PARTNER-CHECK = *STD
Disables the expanded sender checking. The transport address of the partner is not checked, even if the expanded sender checking is globally enabled (see the MODIFY-FT-OPTIONS command).

PARTNER-CHECK = *TRANSPORT-ADDRESS
Enables the expanded sender checking. The transport address is checked, even if the expanded sender checking is globally disabled (see the MODIFY-FT-OPTIONS command). If the transport address under which the partner is reporting does not correspond to the entry in the partner list, the request is rejected.

TRACE =
This option allows you to modify the global settings for partner selection in the *open*FT trace function on a partner-specific basis.

TRACE = *BY-FT-OPTIONS
The global settings apply for the partner.

TRACE = *ON
The trace function is activated for this partner. However, the trace is only written if the global *open*FT trace function is also activated (see also the MODIFY-FT-OPTIONS command, TRACE option).

TRACE = *OFF
The trace function is deactivated for this partner.

AUTH-MANDATORY =
This option allows you to force the authentication of a partner.

AUTH-MANDATORY = *NO
Authentication is not forced, i.e. this partner is not restricted with regard to authentication.

AUTH-MANDATORY = *YES
Authentication is forced, i.e. this partner may only work with authentication.

If the ADD-FT-PARNER command is executed correctly then no message is output.

Command return codes

(SC2)	SC1	Maincode	Meaning
83	32	CMD0221	Internal error.
35	64	FTR1035	User not authorized for this command.
43	64	FTR1043	Partner with same attribute already exists in partner list.
44	64	FTR1044	Maximum number of partners exceeded.

SC1/2 = Subcode 1/2 in decimal notation
For additional information, see section “Command return codes” on page 102

4.7.1 Notes on entering partner systems

The following principles must be adhered to when entering *open*FT partner systems in the partner list:

– Partner systems are principally addressed via the instance IDs of their *open*FT instance (the IDENTIFICATION parameter).

– For partners using *open*FT as of version 8.1, the instance ID is set by the FT administrator of the partner system. Refer to the note in the section “Authentication” on page 46.

– For partners using *open*FT version 8.0 (or earlier), the instance ID has the following format:

  %.<processor>.<entity>

  This enters the address of the main station of the partner system just as it was defined in the partner system or as it was assigned to the partner system by the network administration.

4.7.2 Sample *open*FT partner entries

1. Partner systems via NEA computer-to-computer connection

A partner system that uses *open*FT V8.1 or later is addressed via its host name and instance ID. These can be obtained from the network administrator or the system administrator of the partner system.

Example 1

A partner system that uses *open*FT V9.0 for BS2000 and whose symbolic name is *FTBS2* is to be entered in the partner list. Its processor name is *VAR1* and the instance ID is *VAR1.FUSINET.AT*. The appropriate command is as follows:

```
/ADD-FT-PARTNER                                  -
/     PARTNER-NAME=FTBS2,                        -
/     PARTNER-ADDRESS=VAR1,                      -
/     IDENTIFICATION='VAR1.FUSINET.AT')
```

For partner systems using *open*FT version 8.0 (or earlier), the instance ID is the result of the specifications for the processor name and the *open*FT main station of the partner system (usually, $FJAM).

Example 2

The UNIX based partner system with the symbolic name *FTUNI2*, on which *open*FT V8.0 is installed, is to be entered in the partner list. The partner system is connected via computer interconnection. Its processor name is *UNIX2*, which is defined in the UNIX system by means of the command fta -p. The corresponding command is:

```
/ADD-FT-PARTNER                                  -
/     PARTNER-NAME=FTUNI2                        -
/     PARTNER-ADDRESS=UNIX2,                     -
/          IDENTIFICATION='%.UNIX2.$FJAM'
```

2. Partner systems via TCP/IP

Example

The partner system FTSIE1 with *open*FT V10.0 for UNIX systems is accessed via TCP/IP with the IP address 123.123.45.67. The FT administrator of the partner system has correspondingly assigned it the instance ID %ip123.123.45.67. The default port number for *open*FT is 1100 (the port number is defined via the operating parameters in the UNIX system, the default value is 1100).

```
/ADD-FT-PARTNER                                    -
/     PARTNER-NAME=FTSIE1,                         -
/     PARTNER-ADDRESS=%ip123.123.45.67,            -
/        IDENTIFICATION='%ip123.123.45.67'
```

3. Partner systems via ISO

If the partner system is connected via ISO, the differences relate solely to the generation of the transport system. The partner entry using ADD-FT-PARTNER occurs as described in the section on NEA interconnections on page 110.

4. Partner systems linked via gateways

If you choose the option of linkage via TRANSIT-SNA, specify the partner address as follows:

– Enter the processor name of the front-end processor (PDN host) (LU name) as the host name.

– Enter SNA as the port number for the SNA connection. Enter the station name of the FJAM-LU as the transport selector. Enter the application name of the partner system's main station as the session selector. This application name is defined during VTAM generation. It must comply with the naming conventions, i.e. it must have the form FJM*ftid*.

For more information on specifying the partner address, see section "Defining partner properties" on page 38. Please refer also to the TRANSIT-SNA manual.

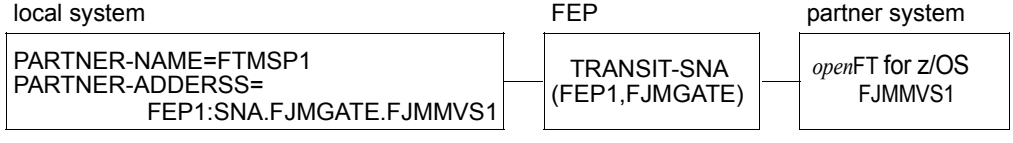
Example 1

The partner system with the symbolic name *FTMSP1*, which is to be linked with the local system via TRANSIT-SNA, is to be entered in the partner list.

The processor name of the FEP is *FEP1*, the station name defined for the FJAM-LU at PDN generation is *FJMGATE*. The remote system's main station was assigned the name *FJMMVS1* at VTAM generation.

The command for entering the partner system in the partner list of *open*FT is as follows:

```
/ADD-FT-PARTNER                                          -
/       PARTNER-NAME = FTMSP1,                           -
/       PARTNER-ADDRESS =FEP1:SNA.FJMGATE.FJMMVS1
```



In the case of an X.25 link, it may also be necessary to retransmit the CUD (call user date). To do this, use a BCMAP command with the L3-CUD parameter.

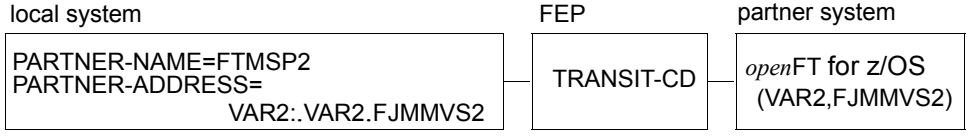
If you select a connection via TRANSIT-CD, then you should use the name generated in BCAM for the processor name of the SNA host as the host name (XSYSP macro for pseudo KOGS). For the transport selector and the session selector, enter the application name which was defined for the partner system's main station during VTAM generation.

Example 2

The partner system with the symbolic name *FTMSP2*, which is to be linked with the local system via TRANSIT-CD, is to be entered in the partner list.

The partner system's main station was assigned the name *FJMMVS2* at VTAM generation. The processor name of the SNA host as defined in BCAM is *VAR2*. The command for entering the partner system is as follows:

```
/ADD-FT-PARTNER                                          -
/       PARTNER-NAME = FTMSP2,                           -
/       PARTNER-ADDRESS = VAR2:.FJMMVS2.FJMMVS2
```



4.7.3 Sample FTAM partner entries

Example 1

The FTAM partner *RITTER* is to be entered in the partner list. At BCAM generation, this system was assigned the processor name *BURGHOF1*. The transport selector is *KUNIBERT*, the session selector is *SESSION1* and the presentation selector is *FTAM*.

```
/ADD-FT-PARTNER RITTER,FTAM://BURGHOF1:.KUNIBERT.SESSION1.FTAM
```

Positional operands were used in this statement which is why the keywords are omitted.

If the partner requires a transport selector which is not in TRANSDATA format (8 character name in EBCDIC, filled with blanks if necessary), this must be defined.

Definition of the FTAM transport selectors in ASCII code:

```
/BCMAP FUNCT=DEFINE,SUBFUNCT=GLOBAL,          -
/      NAME=KUNIBERT,                         -
/      ES=BURGHOF1,                           -
/      PTSEL-I=(4,X‘4654414D‘)
```

Example 2

Since some FTAM implementations respond with another address during connection setup, *open*FT for BS2000 requires a further entry defining the sender address of the partner for the purpose of checking the sender for this partner.

The partner responds with the FTAM1 transport selector, the SESSION2 session selector and the FTAM presentation selector (all in ASCII code):

```
/ADD-FT-PARTNER                                                          -
/    PARTNER-NAME=RITTERXX,                                              -
/    PARTNER-ADDRESS=FTAM://BURGHOF1:.X‘4654414D31404040‘.SESSION2.FTAM,-
/    STATE = *DEACT
```

The relevant BCMAP command must be:

```
/BCMAP FUNCT=DEFINE,SUBFUNCT=GLOBAL,         -
/      NAME=KUNI,                            -
/      ES=BURGHOF1,                          -
/      PTSEL-I=(5,X‘4654414D31‘)
```

Example 3

FTAM link: *open*FT for BS2000 <-> *open*FT for Windows

The FTAM partner WINDOWS is to be entered in the partner list. At BCAM generation, this system was assigned the processor name WINDOWS1. The transport selector is SNI-FTAM in ASCII code and the port number 4800.

```
/ADD-FT-PARTNER WINDOWS,FTAM://WINDOWS1:.SNI-FTAM
```

The relevant BCMAP command must be:

```
/BCMAP FUNCT = DEFINE,SUBFUNCT = GLOBAL,         -
/      NAME = SNI-FTAM,                          -
/      ES = WINDOWS1,                            -
/      PTSEL-I = (8,X‘534E492D4654414D‘),        -
/      PPORT# = 4800
```

4.7.4 Partner systems linked via *open*FTIF

The remote system may be any system that can be accessed via *open*FT for BS2000. There are two possible types of system:

– the target system is an *open*FT partner

– the target system is an FTAM partner

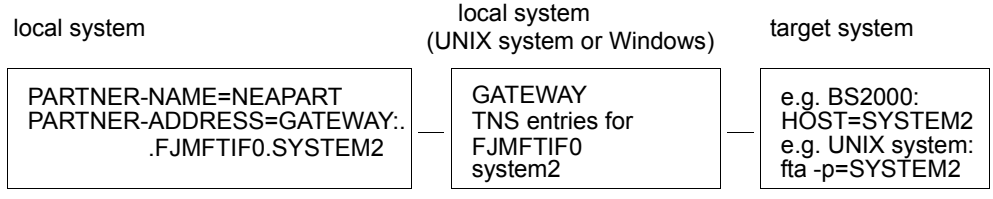
*open*FT partner as target system

You should specify the processor name of the gateway computer as the host name and FJMFTIF0 as the transport selector. As session selector, you should specify the name which is defined for the target system in the gateway computer's transport name server (TNS) (this name must be defined in the gateway computer using lowercase letters).

Example

For the partner system SYSTEM2 using *open*FT V8.0 (or earlier), an entry for an FTIF coupling via the GATEWAY computer could look like this:

```
/ADD-FT-PARTNER                                              -
/       PARTNER-NAME=NEAPART,                                -
/       PARTNER-ADDRESS=GATEWAY:.FJMFTIF0.SYSTEM2
```



If an *open*FT version as of 8.1 is used in the SYSTEM2 partner system, and if the FT administrator of the partner system has specified the instance ID 'system2.fusinet.at', the entry would look like this:

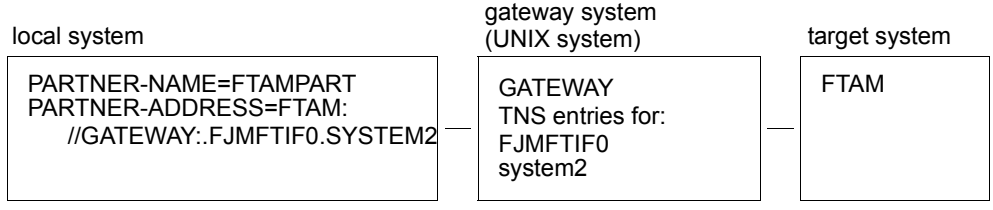
```
/ADD-FT-PARTNER                                              -
/       PARTNER-NAME = NEAPART,                              -
/       PARTNER-ADDRESS=GATEWAY:.FJMFTIF0.SYSTEM2            -
/       IDENTIFICATION = 'system2.fusinet.at',               -
```

FTAM partner as target system

You should specify the processor name of the gateway computer as the host name and FJMFTIF0 as the transport selector. As session selector, you should specify the name which is defined for the target system in the gateway computer's transport name server (TNS). This name is **always** an ASCII name.

Example

```
/ADD-FT-PARTNER                                          -
/    PARTNER-NAME = FTAMPART,                            -
/    PARTNER-ADDRESS = FTAM://GATEWAY:.FJMFTIF0.SYSTEM2
```



4.8 CANCEL-FILE-TRANSFER or CNFT/NCANCEL/FTCANREQ Cancel FT requests

The CANCEL-FILE-TRANSFER (CNFT or NCANCEL or FTCANREQ) command can be used to cancel a file transfer request or to abort the file transfer. The FT system deletes from the request queue the file transfer request that corresponds to the specified selection criteria and, if necessary, aborts the associated file transfer.

The following features apply to this command:

– FT requests submitted either in the local or the remote system can be canceled.

– a single command can be used to cancel several FT requests simultaneously.

– the FT requests to be canceled can be selected using different selection criteria.

You can cancel requests from any user, whereas an FT user can only cancel those FT requests that he/she owns.

The owner of requests issued in the local system is the user ID under which they are submitted.

The owner of requests issued in the remote system is the user ID in the local system under which the requests are executed.

When deleting a file transfer request only values actually specified in the FT request can be used as selection criteria.

Requests which were interrupted and which are waiting to be restarted will block access to the files in question during this phase. Note, however, that this does not apply to library members and POSIX files.

You can fully and unconditionally cancel a selected request and remove it from the request file. “Unconditional” means that, if necessary, the request can be cancelled without any negotiation with the corresponding partner system. In this way, you can clear the request file of requests which are no longer recognized in the partner system or for which there is no longer any connection to the partner system.

WARNING!

If not used carefully, this function can result in inconsistencies in the request files at the corresponding partner systems. Under certain circumstances these inconsistencies may cause baffling error messages (SYSTEM ERROR) and “dead requests” in the partner system request files. It should therefore only be used in exceptional circumstances and after a suitable period has elapsed.

CANCEL-**FILE**-**T**RANSFER / **CNFT** / **NCAN**CEL / **FTCANREQ**

TRANSFER-ID = ***ALL** / <integer 1..2147483639> (**FORCE**-**CAN**CELLATION=***NO**/***YES**)
,**SEL**ECT = ***OWN** / ***PAR**AMETERS(...)
***PAR**AMETERS(...)
OWNER-IDENTIFICATION = ***OWN** / ***ALL** / <name 1..8>
,**INIT**IATOR = (***LOC**AL, ***REMOTE**) / list-poss(2): ***LOC**AL / ***REMOTE**
,**PART**NER = ***ALL** / <text 1..200 with-low>
,**FILE** = ***ALL** / <filename 1..54> /<c-string 1..512 with-low> / ***LIB**RARY-**ELEM**ENT(...) /
***POS**IX(**NAME** = <posix-pathname 1..219>) / ***PUBSET**(**PUBSET** = <cat-id 1..4>)
***LIB**RARY-**ELEM**ENT(...)
LIBRARY = ***ALL** / <filename 1..54>
,**ELEM**ENT = ***ALL** / <filename 1..64 without-gen-vers>(...) /
<composed-name 1..64 with-under>(...)
<filename>(...) / <composed-name>(...)
VERSION = ***ALL** / <text 1..24>
,**TYPE** = ***ALL** / <name 1..8>
,**MONJV** = ***NONE** / <filename 1..54 without-gen-vers>
,**JV-PASS**WORD = ***NONE** / <c-string 1..4> / <x-string 1..8> /
<integer -2147483648..2147483647> / ***SECRET**

Operands

TRANSFER-ID =
Transfer ID of the FT request to be canceled.

TRANSFER-ID = *ALL
Deletes all FT requests if no further selection criteria are specified with SELECT. FT administrators can delete all current FT requests that access the system. FT users can only delete FT requests of their own ID using this entry.

TRANSFER-ID = <integer 1..2147483639>
Request identification which was communicated to the local system in the FT request confirmation. The associated **FORCE-CANCELLATION** parameter is available only to the FT administrator. It is used for an unconditional request cancellation.

TRANSFER-ID = <integer 1..2147483639>(FORCE-CANCELLATION=*NO)
NO is the default value. The request is removed from the request file following negotiation with the partner system.

TRANSFER-ID = <integer 1..2147483639>(FORCE-CANCELLATION=*YES)
The request is removed from the request file without negotiation with the partner system. For this to be possible, you must possess the FT-ADM privilege and the request must previously have been cancelled with CANCEL-FILE-TRANSFER **FORCE=*NO.**

SELECT =
Contains selection criteria for FT requests to be canceled. A request is canceled if it satisfies all the specified criteria.

SELECT = *OWN
Cancels all FT requests associated with the ID of the user.

SELECT = *PARAMETERS(...)

OWNER-IDENTIFICATION =
Designates the owner of the FT requests.

OWNER-IDENTIFICATION = *OWN
Cancels only the FT requests under the user's own ID.

OWNER-IDENTIFICATION = *ALL
Cancels FT requests under all user IDs. Only the administrator can use this entry.

OWNER-IDENTIFICATION = <name 1..8>
Specifies a particular user ID whose FT requests are to be canceled.

INITIATOR =
Specifies the initiator of the FT requests to be canceled.

INITIATOR = (*LOCAL,*REMOTE)
Cancels FT requests in the local system and in remote systems.

INITIATOR = *LOCAL
Cancels FT requests issued in the local system.

INITIATOR = *REMOTE
Cancels FT requests issued in remote systems.

PARTNER =
Cancels FT requests that were to be executed with a specific partner system.

PARTNER = *ALL
The name of the partner system is not used as a selection criterion to determine the FT requests to be canceled.

PARTNER = <text 1..200 with-low>
The FT requests that were to be executed with this partner are to be canceled. You can specify either the name of the partner system from the partner list or the address of the partner system.

FILE =
Cancels all FT requests in the local system that access this file, this pubset or this library element whether as a send file or receive file. The file name or library member name must be specified exactly as it appears in the file transfer request.

FILE = *ALL
The file name is not used as a selection criterion to determine the FT requests to be canceled.

FILE = <filename 1..54> / <c-string 1..221 with-low>/
***POSIX(NAME=<posix-pathname 1..219>)**
Cancels FT requests in the local system that access this file.

FILE = *PUBSET(PUBSET = <cat-id 1..4>)
Deletes all FT requests that have locked files on the specified pubset.

FILE = *LIBRARY-ELEMENT(...)
Cancels FT requests that access library members in the local system. The parameters in parentheses can also be used as positional operands without specifying the associated keywords.

LIBRARY =
Selects the library concerned.

LIBRARY = *ALL
The library name is not used as a selection criterion to determine the FT requests to be canceled.

LIBRARY = <filename 1..54>
FT requests that access this library are to be canceled.

ELEMENT =
Selects the library concerned.

ELEMENT = *ALL
The name of the library member is not a selection criterion to determine the FT requests to be canceled.

ELEMENT = <filename 1..64 without-gen-vers>(...) /
<composed-name 1..64 with-under>(...)
Is the name of the library member concerned.

VERSION =
Specifies the version of the library member.

VERSION = *ALL
The version of the library member is not a selection criterion for the FT requests to be canceled.

VERSION = <text 1..24>
Only FT requests that access this version of the library member are to be canceled.

TYPE =
specifies the type of the library member concerned.

TYPE = *ALL
The type of library member is not used as a selection criterion to determine the FT requests to be canceled.

TYPE = <name 1..8>
Only FT requests that access library members of this type are to be canceled.

MONJV =
If appropriate, selects the specific FT request that is being monitored by this job variable.

MONJV = *NONE
A job variable is not used as a selection criterion to cancel the file transfer.

MONJV = <filename 1..54 without-gen-vers>
The FT monitored by this job variable is to be canceled.

JV-PASSWORD =
If required, specifies the password needed to access the job variable.
If you have already notified the system of the password with the BS2000 command ADD-PASSWORD, you do not have to specify JV-PASSWORD.

JV-PASSWORD = *NONE
The job variable is not password-protected.

JV-PASSWORD = <c-string 1..4> / <x-string 1..8> /
<integer -2147483648..2147483647>
This password is required to access the job variable.

JV-PASSWORD = *SECRET
The system issues the request to enter the password. However, input is not displayed on the screen.

The specification of more than one selection criteria in the CANCEL-FILE-TRANSFER command may result in a file transfer request being “overdefined” (e.g. by entries for TRANSFER-ID and MONJV). If in such a case the specified criteria conflict, the CANCEL-FILE-TRANSFER command is acknowledged with the following message:

```
%  FTR0504 No requests available for the selection criteria
```

In such a case there is no jump to the next SET-JOB-STEP in procedures.

Command return codes

(SC2)	SC1	Maincode	Meaning
0	0	CMD0001	There are no requests that meet the specified selection criteria.
32	32	CMD0221	Request rejected. Internal error. Job variable not accessible.
33	32	CMD0221	Request rejected. Internal error.
36	32	CMD0221	Request rejected. Request data inconsistent.
82	32	CMD0221	Internal error. Job variable not accessible.
83	32	CMD0221	Internal error.
36	64	FTR1036	User not authorized for other user IDs.
38	64	FTR1038	Request is in the termination phase and can no longer be cancelled.
47	64	FTR1047	Request with the specified transfer ID could not be found.
226	64	FTR2226	Job variable contents inconsistent
227	64	FTR2227	Job variable not in use by *open*FT.
228	64	FTR2228	Job variable not found.

SC1/2 = Subcode 1/2 in decimal notation
For additional information, see section “Command return codes” on page 102

Example

If more than one request is canceled using a CANCEL-FILE-TRANSFER command, the following query is issued:

```
% FTR0560 Cancel all specified requests? Reply (y=yes; n=no)
```

The cancel request can be rescinded using `NO`.

4.9 CREATE-FT-INSTANCE Create a new *open*FT instance or activate an unloaded *open*FT instance

Using this command, you create a new administration entry for an instance and load the instance. You can optionally create the instance in such a manner that when the subsystem FT is started, *open*FT is also automatically started in this instance (i.e. a START-FT command is no longer necessary). In addition, the command re-activates or reloads an instance that was unloaded using DELETE-FT-INSTANCE.

CREATE-**FT**-**INST**ANCE
NAME = <alphanum-name 1...8> **,CON**FIG-**USER**ID = <text 1..15> **,AUTO**MATIC-**START** = ***OFF** / ***ON**

Operands

NAME = <alphanum-name 1..8>
The name of the *open*FT instance that is to be created. This name must be identical on all of the computers on which this instance is to be used.

CONFIG-USERID = <text 1..15>
The file name prefix of the *open*FT instance variable files. The prefix must consist of a catalog name and a USER-ID. This USER-ID is designated as the configuration user ID of the instance.

AUTOMATIC-START =
This is specified if an automatic start of *open*FT is to occur within an instance, after loading the instance.

AUTOMATIC-START = *OFF
*open*FT is not started after loading the instance .

AUTOMATIC-START = *ON
After each loading of the instance, a START-FT command is implicitly executed in this instance. By doing this, it is possible to immediately work with *open*FT after loading. All the components which are available to a standard instance are also started, such as, for example *open*FT-AC, *open*FT-FTAM and *open*FT-FTP.

Command return codes

(SC2)	SC1	Maincode	Meaning
195	1	CMD0202	Invalid parameter
83	32	CMD0221	Internal error.
22	64	FTR1022	Instance already exists
23	64	FTR1023	Maximum number of instances exceeded
2	1	FTR1028	Config user ID not accessible

SC1/2 = Subcode 1/2 in decimal notation

For additional information, see section “Command return codes” on page 102

4.10 CREATE-FT-KEY-SET (FTCREKEY) Create a key pair set

Using this command, you create a key pair for authenticating your *open*FT instance in partner systems (RSA procedures). The key pair consists of a private key, administered internally by *open*FT, and a public key. Public keys are stored on the configuration user ID of the *open*FT instance (default: $SYSFJAM) under the name **SYSPKF.R<key reference>.L<key length>**. The key reference is a numerical designator for the version of the key pair. The key length is 768 or 1024. The two key lengths are always generated. The public key files are text files which are created in the character code of the respective operating system, i.e. EBCDIC.DF04-1 for BS2000 and z/OS, ISO8859-1 for UNIX systems and CP1252 for Windows systems.

In a SYSPKF.COMMENT file, you can store comments written in the first lines of the public key files when a key pair set is created on the configuration user ID of the *open*FT instance. Such comments could be, for example, the communications partner and the telephone number of the FT administrator on duty. The lines in the SYSPKF.COMMENT file may be a maximum of 78 characters long.

So that your *open*FT instance can be authenticated by partner systems (using *open*FT as of version 8.1), the public key file must be transported to the partners via a reliable path and re-coded if necessary (see section “Authentication” on page 46).

In order to make an authorized update of the key pair sets, *open*FT supports up to three key pair sets at a time.

The most current key pair is used for delivering the session key for encrypting user data and request description data. If there is no key pair set, work proceeds without encryption.

CREATE-**FT-KEY**-SET/ **FTCREKEY**

Without operands

Command return codes

(SC2)	SC1	Maincode	Meaning
83	32	CMD0221	Internal error.
87	32	CMD0221	No space left on device for internal files.
29	64	FTR1029	Maximum number of key pairs exceeded.
35	64	FTR1035	User not authorized for this command.

SC1/2 = Subcode 1/2 in decimal notation

For additional information, see section “Command return codes” on page 102

4.11 CREATE-FT-PROFILE Create admission profile

A prerequisite for using this command is the use of *open*FT-AC.

The FTAC administrator can use CREATE-FT-PROFILE to create admission profiles for each user. Admission profiles pre-defined by the FTAC administrator must be activated by the user using MODIFY-FT-PROFILE (see page 199ff) before they can be used. In addition, FTAC users can use CREATE-FT-PROFILE to create admission profiles on their own user IDs. Profiles created by the FTAC administrator can be used immediately if the FTAC administrator also possesses the TSOS privilege.

CREATE-**FT-PROF**ILE

NAME = <alphanum-name 1..8>

,**PASS**WORD = ***NONE** / <c-string 1..8 with-low> / <x-string 1..16> / ***SEC**RET

,**TRANS**FER-**AD**MISSION = ***NOT-SPEC**IFIED / <alphanum-name 8..32>(...) /
<c-string 8..32 with-low>(...) / <x-string 15..64>(...) / ***SEC**RET

<alphanum-name 8..32>(...) / <c-string 8..32 with-low>(...) / <x-string 15..64>(...)

VALID = ***YES** / ***NO**

,**USAGE** = ***PRIVATE** / ***PUB**LIC

,**EXP**IRATION-**DATE** = ***NOT-REST**RICTED / <date 8..10>

,**PRIV**ILEGED = ***NO** / ***YES**

,**IGN**ORE-**MAX-LEV**ELS = ***NO** / ***YES** / ***PAR**AMETERS(...)

***PAR**AMETERS(...)

OUTBOUND-**SEND** = ***NO** / ***YES**

,**OUT**BOUND-**REC**EIVE = ***NO** / ***YES**

,**IN**BOUND-**SEND** = ***NO** / ***YES**

,**IN**BOUND-**REC**EIVE = ***NO** / ***YES**

,**IN**BOUND-**PROC**ESSING = ***NO** / ***YES**

,**IN**BOUND-**MAN**AGEMENT = ***NO** / ***YES**

,**USER-AD**MISSION = ***OWN** / ***PAR**AMETERS(...)

***PAR**AMETERS(...)

USER-IDENTIFICATION = ***OWN** / <name 1..8>

,**ACCOUNT** = ***OWN** / ***FIRST**/ ***NOT-SPEC**IFIED / <alphanum-name 1..8>

,**PASS**WORD = ***OWN** / ***NOT-SPEC**IFIED / <c-string 1..8> / <c-string 9..32> / <x-string 1..16> /
***NONE** / ***SEC**RET

,**INIT**IATOR = (***LOC**AL, ***REM**OTE) / list-poss(2): ***LOC**AL / ***REM**OTE /

,**TRANS**FER-**DIR**ECTION = ***NOT-REST**RICTED / **FROM**-PARTNER / **TO**-PARTNER

,**PART**NER = ***NOT-REST**RICTED / list-poss(50): <text 1..200 with-low>

```
,MAX-PARTNER-LEVEL = *NOT-RESTRICTED / <integer 0..100>
,FILE-NAME = *NOT-RESTRICTED / <filename1..54> / <c-string 1..512 with-low> /
                *EXPANSION(...) / *LIBRARY-ELEMENT(...) / *POSIX(NAME=<posix-pathname 1..219>)
   *EXPANSION(...)
      PREFIX = <filename 1..53> / <partial-filename 2..53> / <c-string 1..511 with-low>
   *LIBRARY-ELEMENT(...)
      LIBRARY = *NOT-RESTRICTED / <filename 1..54> / *EXPANSION(...)
         *EXPANSION(...)
            PREFIX = <filename 1..53> / <partial-filename 2..53>
      ,ELEMENT = *NOT-RESTRICTED / <composed-name 1..64 with-under>(...) / *EXPANSION(...)
         <composed-name 1..64 with-under>(...)
            VERSION = *STD / <text 1..24>
         *EXPANSION(...)
            PREFIX = <composed-name 1..63 with-under> / <partial-filename 2..63>
      ,TYPE = *NOT-RESTRICTED / <name 1..8>
,FILE-PASSWORD = *NOT-RESTRICTED / *NONE / <c-string 1...4> / <x-string 1...8> /
                    <integer -2147483648...2147483647> / *SECRET
,PROCESSING-ADMISSION = *SAME / *NOT-RESTRICTED / *PARAMETERS(...)
   *PARAMETERS(...)
      USER-IDENTIFICATION = *SAME / *NOT-RESTRICTED / <name 1..8>
      ,ACCOUNT = *SAME / *NOT-RESTRICTED / <alphanum-name 1..8>
      ,PASSWORD = *SAME / *NOT-RESTRICTED / *NONE / <c-string 1..8> / <c-string 9..32> /
                    <x-string 1..16> / *SECRET
,SUCCESS-PROCESSING = *NOT-RESTRICTED / *NONE / <c-string 1..1000 with-low> /
                          *EXPANSION(...)
   *EXPANSION(...)
      PREFIX = *NOT-RESTRICTED / <c-string 1..999 with-low>
      ,SUFFIX = *NOT-RESTRICTED / <c-string 1..999 with-low>
,FAILURE-PROCESSING = *NOT-RESTRICTED / *NONE / <c-string 1..1000 with-low> /
                          *EXPANSION(...)
   *EXPANSION(...)
      PREFIX = *NOT-RESTRICTED / <c-string  1..999 with-low>
      ,SUFFIX = *NOT-RESTRICTED / <c-string 1..999 with-low>
,WRITE-MODE = *NOT-RESTRICTED / *NEW-FILE / *REPLACE-FILE / *EXTEND-FILE
,FT-FUNCTION = *NOT-RESTRICTED / list-poss(4): *TRANSFER-FILE / *MODIFY-FILE-ATTRIBUTES /
                    *READ-DIRECTORY /*FILE-PROCESSING
,USER-INFORMATION = *NONE / <c-string 1..100 with-low>
,DATA-ENCRYPTION = *NOT-RESTRICTED / *NO / *YES
```

Operands

NAME = <alphanum-name 1..8>
With NAME, the admission profile is given a name. This name must be unique among all admission profiles on that user ID. If an admission profile with this name already exists, FTAC rejects the command with the message:

```
FTC0100 FT profile already exists.
```

The command SHOW-FT-PROFILE (see page 279ff) can be used to view the already existing names. To obtain this information, the command SHOW-FT-PROFILE can be entered without operands .

PASSWORD =
With PASSWORD, you enter the FTAC password which authorizes you to issue FTAC commands on your user ID, if such a password was defined in your admission set.

PASSWORD = *NONE
No FTAC password is required.

PASSWORD = <c-string 1..8 with-low> / <x-string 1..16>
This FTAC password is required.

PASSWORD = *SECRET
The system prompts you to input the password. However, the password does not appear on the screen.

TRANSFER-ADMISSION =
With TRANSFER-ADMISSION, you define transfer admission. If this transfer admission is entered in an FT request instead of the LOGON admission, then the access rights are valid which are defined in this admission profile. This transfer admission must be unique in the entire *open*FT system, so that there is no conflict with other transfer admissions which other FTAC users have defined for other access rights. When the transfer admission which you have selected has already been used, then FTAC rejects the command with the message:

```
FTC0101 Transfer admission already exists.
```

The FTAC administrator can also assign a transfer admission when he creates an admission profile for a user ID. To do this, the complete USER-ADMISSION must be entered for the user ID in question (USER-IDENTIFICTATION, ACCOUNT and PASSWORD).

TRANSFER-ADMISSION = *NOT-SPECIFIED
This entry is used to set up a profile without transfer admission. Such a profile remains inaccessible until a valid transfer admission has been specified.

TRANSFER-ADMISSION = <alphanum-name 8..32>(...) /
<c-string 8..32 with-low>(...) / <x-string 15..64>(...)

The character string must be entered as the transfer admission in the transfer request. The alphanumeric entry is always stored in lower-case letters.

VALID = *YES
The transfer admission is valid.

VALID = *NO
The transfer admission is not valid. With this entry, users can be denied access to the profile.

USAGE = *PRIVATE
Access to your profile is denied for security reasons, when someone with another user ID attempts a second time to specify the TRANSFER ADMISSION which has already been used by you.

USAGE = *PUBLIC
Access to your profile is not denied if another user happens to “discover” your TRANSFER-ADMISSION. “Discovery” means that another user ID attempted to specify the same TRANSFER ADMISSION twice. This is rejected for security reasons.

EXPIRATION-DATE = *NOT-RESTRICTED
The use of this transfer admission is not restricted with respect to time.

EXPIRATION-DATE = <date 8..10>
The use of the transfer admission is only possible until the given date (max. 19.01. 2038). The entry must be made in the form *yyyy-mm-dd* or *yy-mm-dd*.

TRANSFER-ADMISSION = *SECRET
The system prompts you to input the transfer admission; however, this does not appear on the screen. The operands VALID, USAGE and EXPIRATION-DATE can also be secretly entered in this case.

PRIVILEGED =
With PRIVILEGED, the FTAC administrator can privilege the profile. FT requests which are processed with a privileged admission profile are not subject to the restrictions which are set for MAX-ADM-LEVEL (see page 245) in the admission set.

PRIVILEGED = *NO
The admission profile is not privileged.

PRIVILEGED = *YES
The admission profile is privileged.
Only the FTAC administrator can use this entry.

IGNORE-MAX-LEVELS =
With IGNORE-MAX-LEVELS, you can determine for which of the six basic functions the restrictions of the admission set should be ignored. The user's MAX-USER-LEVELS can be exceeded in this way. The MAX-ADM-LEVELS in the admission set can only be effectively exceeded with an admission profile which has been designated as privileged by the FTAC administrator. The FTAC user can set up an admission profile for himself for special tasks (e.g. sending a certain file to a partner system with which he normally is not allowed to conduct a file transfer), which allows him to exceed the admission set. This profile must be explicitly given privileged status by the FTAC administrator.
If you enter IGNORE-MAX-LEVELS=*YES, the settings for **all** the basic functions are ignored. If you wish to ignore the admission set for **specific** basic functions, you need to do this with the operands explained later in the text.
The following table shows which partial components of the file management can be used under which conditions:

Inbound file management function	**Setting in admission set/extension in profile**
Show file attributes	Inbound sending (IBS) permitted
Modify file attributes	Inbound receiving (IBR) **and** Inbound file management (IBF) permitted
Rename files	Inbound receiving (IBR) **and** Inbound file management (IBF) permitted
Delete files	Inbound receiving (IBR) permitted **and** write rule = overwrite in profile
Show directories	Inbound file management (IBF) permitted **and** direction = to partner in profile
Create, rename, delete directories	Inbound file management (IBF) permitted **and** direction = from partner in profile

IGNORE-MAX-LEVELS = *NO
FT requests which are processed with the admission profile are subject to the restrictions of the admission set.

IGNORE-MAX-LEVELS = *YES
*YES allows you to communicate with partner systems whose security level exceeds the specifications of the admission set. Unless you have a privileged profile, you can only exceed the MAX-USER-LEVELS and not the MAX-ADM-LEVELS in the admission set. You must respect the restrictions defined in the admission set by the FTAC administrator. The SHOW-FT-ADMISSION-SET command provides information on the entries made by the FTAC administrator. This includes information about the current MAX-USER-LEVELS and MAX-ADM-LEVELS settings.

IGNORE-MAX-LEVELS = *PARAMETERS(...)
The following operands can be used to selectively deactivate the default settings for the individual basic functions.

OUTBOUND-SEND = *NO
The maximum security level which can be reached with the basic function “outbound send” is determined by the admission set.

OUTBOUND-SEND = *YES
For the basic function “outbound send”, you can use this admission profile to disregard the MAX-USER-LEVELS. If your profile is privileged, you are also not held to the restrictions of the MAX-ADM-LEVELS.

OUTBOUND-RECEIVE = *NO
The maximum security level which can be reached with the basic function “outbound receive” is determined by the admission set.

OUTBOUND-RECEIVE = *YES
For the basic function “outbound receive”, you can use this admission profile to disregard the MAX-USER-LEVELS. If your profile is privileged, you are also not held to the restrictions of the MAX-ADM-LEVELS.

INBOUND-SEND = *NO
The maximum security level which can be reached with the basic function “inbound send” is determined by the admission set.

INBOUND-SEND = *YES
For the basic function “inbound send”, you can use this admission profile to disregard the MAX-USER-LEVELS. If your profile is privileged, you are also not held to the restrictions of the MAX-ADM-LEVELS. In addition, the partial component “display file attributes” of the basic function “inbound file management” can be used.

INBOUND-RECEIVE = *NO
The maximum security level which can be reached with the basic function “inbound receive” is determined by the admission set.

INBOUND-RECEIVE = *YES
With this profile, you can disregard your settings for “inbound receive” in the MAX-USER-LEVELS. If your profile is privileged, you are also not held to the restrictions of the MAX-ADM-LEVELS. In addition, the following partial components of the basic function “inbound file management” can be used:
– delete files, as long as the file attributes are set accordingly,
– modify file attributes, if the basic function “inbound file management” was admitted in the admission set or in the admission profile.

INBOUND-PROCESSING = *NO
The maximum security level which can be reached with the basic function “inbound follow-up processing” is determined by the admission set.

INBOUND-PROCESSING = *YES
For the basic function “inbound follow-up processing”, you can use this admission profile to disregard the MAX-USER-LEVELS. If your profile is privileged, you are also not held to the restrictions of the MAX-ADM-LEVELS.

INBOUND-MANAGEMENT = *NO
The maximum security level which can be reached with the basic function “inbound file management” is determined by the admission set.

INBOUND-MANAGEMENT = *YES
For the basic function “inbound file management”, you can use this admission profile to disregard the MAX-USER-LEVELS. If your profile is privileged, you are also not held to the restrictions of the MAX-ADM-LEVELS. The partial component “modify file attributes” of the basic function “inbound file management” only functions if the basic function “inbound receive” was admitted in the admission set or admission profile.

USER-ADMISSION =
With USER-ADMISSION, the user specifies the user ID under which the profile is to be saved. FT requests which work with this admission profile access the given user ID in the local system.

If, as FTAC administrator, you create the admission profile for a user, you cannot generally specify either ACCOUNT or PASSWORD in the USER-ADMISSION operand (since these should be known only to the user in question). These specifications must be entered by the user by means of MODIFY-FT-PROFILE before the profile can actually be used. If, as FTAC administrator, you also possess the TSOS privilege, then you can also create a profile which is available for immediate use even without a password specification.

If you want to assign a transfer admission to a user’s admission profile then you must specify both the USER-ADMISSION and the ACCOUNT and PASSWORD.

USER-ADMISSION = *OWN
For USER-IDENTIFICATION and ACCOUNT, the specifications for your user ID and your account number are taken from your LOGON authorization. A BS2000 password is only taken from your LOGON authorization when an FT request accesses the admission profile.

USER-ADMISSION = *PARAMETERS(...)
You can also enter the individual components of the user ID. This allows you to keep FT requests which use this admission profile under a different account number, for example. Or, a password can be set in the admission profile. FT requests which use this admission profile will then only function if their current LOGON password corresponds to the pre-set password.

USER IDENTIFICATION =
With USER-IDENTIFICATION, you enter your user ID in BS2000.

USER-IDENTIFICATION = *OWN
Your user ID is taken from the LOGON authorization.

USER-IDENTIFICATION = <name 1..8>
This is the user ID associated with the profile.

ACCOUNT =
With ACCOUNT, you enter your account number under which an FT request is to be kept when it uses this admission profile.

ACCOUNT = *NOT-SPECIFIED
The account number is first entered by the owner of the admission profile. This function allows the FTAC administrator to create profiles for foreign user IDs.

ACCOUNT = *OWN
The account number is taken from your LOGON authorization.

ACCOUNT = *FIRST
The first account number assigned to the home pubset of the specified USER-IDENTIFICATION at the time the profile is used in the system. Used for account assignment in the case of transfer requests. If the ID's account number changes, the profile no longer has to be modified as was the case in the past.

ACCOUNT = <alphanum-name 1..8>
An FT request should be kept under the account number specified when it accesses this admission profile. You can enter any account number which is associated with the user ID.

PASSWORD =
With PASSWORD, you enter the BS2000 password associated with your user ID.

PASSWORD = *OWN
When an FT request refers to this admission profile, FTAC uses the BS2000 password valid for your user ID at that moment. This prevents you from having to modify the admission profile if the BS2000 password is changed.

PASSWORD = *NONE
No BS2000 password is required for the user ID.

PASSWORD = *NOT-SPECIFIED
The password will be entered by the owner of the admission profile. This function allows the FTAC administrator to create profiles for foreign user IDs.

PASSWORD = <c-string 1..8> / <c-string 9..32> / <x-string 1..16>
When an FT request accesses the admission profile, the password specified is compared with the current LOGON password. If the two do not correspond, the FT request is rejected.

PASSWORD = *SECRET
The system prompts you to enter the password. The entry does not appear on the screen.

INITIATOR =
With INITIATOR, you determine if initiators from local and/or remote systems are permitted to use this admission profile for their FT requests.

INITIATOR = (*LOCAL,*REMOTE)
This admission profile may be used by initiators from local and remote systems.

INITIATOR = *REMOTE
This admission profile may only be used for FT requests by initiators from remote systems.

INITIATOR = *LOCAL
This admission profile may only be used for FT requests by initiators from the local system.

TRANSFER-DIRECTION =
With TRANSFER-DIRECTION, you determine which transfer direction may be used with this admission profile. The transfer direction is always determined from the system in which the admission profile was defined.

TRANSFER-DIRECTION = *NOT-RESTRICTED
With this admission profile, files can be transferred to and from a partner system.

TRANSFER-DIRECTION = *FROM-PARTNER
With this admission profile, files can only be transferred from a partner system to your system. It is not possible to display file attributes/directories (partial components of “inbound file management”).

TRANSFER-DIRECTION = *TO-PARTNER
With this admission profile, files can only be transferred from your system to a partner system. It is not possible to modify file attributes or delete files (partial components of “inbound file management”).

PARTNER =
With PARTNER, you can specify that this admission profile is to be used only for FT requests which are processed by a a certain partner system.

PARTNER = *NOT-RESTRICTED
The range of use for this admission profile is not restricted to FT requests with certain partner systems.

PARTNER = list-poss(50): <text 1..200 with-low>
The admission profile only permits those FT requests which are processed with the specified partner systems. A maximum of 50 partner names can be specified. The total length of all the partners may not exceed 1000 characters. You may specify the name from the partner list or the fully qualified address of the partner system, see also section “Specifying partner addresses” on page 39. It is advisable to use the name from the partner list.

MAX-PARTNER-LEVEL =
With MAX-PARTNER-LEVEL, a maximum security level can be specified. The admission profile will then only permit those FT requests which are processed with partner systems which have this security level or lower.
MAX-PARTNER-LEVEL works in conjunction with the admission set. When non-privileged admission profiles are used, the access check is executed on the basis of the smallest specified value.

MAX-PARTNER-LEVEL = *NOT-RESTRICTED
If FT requests are processed with this admission profile, then the highest accessible security level is determined by the admission set.

MAX-PARTNER-LEVEL = <integer 0..100>
With this admission profile, all partner systems which have this security level or lower can be communicated with.

When you set MAX-PARTNER-LEVEL=0, you prevent access to the admission profile (for the moment). No FT requests can be processed with this admission profile.

FILE-NAME =
With FILE-NAME, you determine which files or library members under your user ID may be accessed by FT requests that use this admission profile.

FILE-NAME = *NOT-RESTRICTED
The admission profile permits unrestricted access to all files and library members of the user ID.

FILE-NAME = <filename 1..54> / <c-string 1..512 with-low>/
***POSIX(NAME = <posix-pathname 1..219>)**

With this admission profile, only the specified file may be accessed.
However, *open*FT is also able to generate unique filenames automatically, thus providing an easy way of avoiding conflicts. This is done by specifying the string %UNIQUE at the end of the filename which is predefined here (see the section “File names” in the User Guide). When follow-up processing is specified, this file can be referenced with %FILENAME.

You can also directly specify file transfer with file pre- or post-processing here by entering a pipe symbol ’|’ followed by a command.

FILE-NAME = *EXPANSION
(PREFIX=<filename 1..53> / <partial-filename 2..53> /
<c-string 1..511 with-low>)

This entry can be used to restrict access to a number of files which all begin with the same prefix. If a filename is entered in an FT request which works with this admission profile, FTAC sets the *prefix* defined with EXPANSION in front of this filename. The FT request is then permitted to access the file *PrefixFilename*.

Example

PREFIX=DONALD.; An FT request in which the FILE-NAME=DUCK is specified accesses the file DONALD.DUCK.

Please note that the part of a DVS filename which is specified in the file transfer command still has to be of the type <filename>.

If you want to perform file transfer with pre- or post-processing, you should indicate this by entering the pipe symbol ‘|’ at the start of the prefix. The created FTAC profile can then be used only for file transfer with pre- or post-processing since the file name that is generated also starts with a ‘|’. The maximum length of the full pre- or post-processing command is restricted to the maximum filename length.

Example

```
FILE-NAME=C‘|/Command1;/Command2;/Command3; ...‘
```

In this case, the characters available for the pre- or post-processing command in the FT request are limited to:

– alphanumeric characters (letters and digits)
– the special characters + = / ! - , @ ␣ , _ ” $ ’
– a period (.) between alphanumeric characters

In the case of admission profiles which are to be used exclusively for the ftexec command you must specify a filename or filename prefix that starts with the character string ‘|ftexecsv’ (see example 2).

FILE-NAME = *LIBRARY-ELEMENT(...)
With *LIBRARY-ELEMENT, you determine which of your libraries and library members may be accessed by FT requests which use this admission profile.

LIBRARY =
With LIBRARY, you define which libraries may be accessed with this admission profile.

LIBRARY = *NOT-RESTRICTED
The admission profile does not restrict access to libraries.

LIBRARY = <filename 1..64>
With this admission profile, only this library may be accessed.

LIBRARY = *EXPANSION
(PREFIX=<filename1..53> / <partial-filename 2..53>)
With this admission profile, only those libraries may be accessed which begin with the specified prefix. FTAC sets the prefix in front of a library name in an FT request which works with this admission profile, and then permits access to the library *Prefix-Libraryname.*

ELEMENT =
With ELEMENT, you determine which library members may be accessed with this admission profile.

ELEMENT = *NOT-RESTRICTED
The admission profile permits unrestricted access to library members.

ELEMENT = <composed-name 1..64 with-under>(...)
The admission profile only permits access to the specified library member.

VERSION =
With VERSION, access is only permitted for a specific version of the library member.

VERSION = *STD
The admission element permits access only to the highest version of the library member.

VERSION = <text 1..24>
Access is only permitted for this version of the library member.

ELEMENT = *EXPANSION (PREFIX=<partial-filename 2..63> /
<composed-name 1..63 with-under>)
With *EXPANSION, you define a prefix. When a name for a library member is specified in an FT request which works with this admission profile, FTAC adds the specified prefix to this member name. The admission profile then permits access to this member with the name *PrefixMembername*.

TYPE =
With TYPE, you can specify a certain type of library member. The admission profile then only permits access to library members of this type.

TYPE = *NOT-RESTRICTED
Access is not restricted to a certain type of library member.

TYPE = <name 1..8>
FT requests which work with this admission profile may only access library members of this type.

FILE-PASSWORD =
With FILE-PASSWORD, you can enter a password for files into the admission profile. The FTAC functionality then only permits access to files which are protected with this password and to unprotected files. When a FILE-PASSWORD is specified in an admission profile, the password may no longer be specified in an FT request which uses this admission profile. This allows you to permit access to certain files to users in remote systems, without having to give away the file passwords.

FILE-PASSWORD = *NOT-RESTRICTED
The admission profile permits access to all files. If a password is set for a file, then it must be specified in the transfer request.

FILE-PASSWORD = *NONE
The admission profile only permits access to files without file passwords.

FILE-PASSWORD = <c-string 1..4> / <x-string 1..8> /
<integer -2147483648..2147483647>
The admission profile only permits access to files which are protected with the password specified and to unprotected files. The password which has already been specified in the profile may not be repeated in the transfer request. PASSWORD=*NONE would be entered in this case!

FILE-PASSWORD = *SECRET
The system prompts you to enter the password. However, the password does not appear on the screen.

PROCESSING-ADMISSION =
With PROCESSING-ADMISSION, you can enter a user ID in your BS2000 system. Any follow-up processing of an FT request will be executed under this user ID. With PROCESSING-ADMISSION in the admission profile, you do not need to disclose your LOGON authorization to partner systems for follow-up processing.

PROCESSING-ADMISSION = *SAME
For the PROCESSING-ADMISSION, the values of the USER-ADMISSION are used. If *SAME is entered here, then any FT request which uses this profile must also contain PROCESSING-ADMISSION=*SAME or PROCESSING-ADMISSION= *NOT-SPECIFIED.

PROCESSING-ADMISSION = *NOT-RESTRICTED
FT requests which use this admission profile may contain any PROCESSING-ADMISSION. If you wish to perform follow-up processing with FTAM partners, PROCESSING-ADMISSION must have a value other than *NOT-RESTRICTED.

PROCESSING-ADMISSION = *PARAMETERS(...)
You can also enter the individual components of the user ID. This allows you to keep FT requests which use this admission profile under a different account number, for example. Or, a password can be set in the admission profile. FT requests which use this admission profile will then only function if their current LOGON password corresponds to the pre-set password.

USER-IDENTIFICATION =
USER-IDENTIFICATION identifies the user ID under which the follow-up processing is to be executed.

USER-IDENTIFICATION = *SAME
The USER-IDENTIFICATION is taken from the USER-ADMISSION.

USER-IDENTIFICATION = *NOT-RESTRICTED
The admission profile does not restrict the user ID for the follow-up processing.

USER-IDENTIFICATION = <name 1..8>
FT requests which are processed with this admission profile are only permitted follow-up processing under this user ID. If another user ID is entered here, the parameter PASSWORD must also be entered. PASSWORD=*SAME is then not valid.

ACCOUNT =
With ACCOUNT, you specify the account number for the follow-up processing.

ACCOUNT = *SAME
The account number is taken from the USER-ADMISSION.

ACCOUNT = *NOT-RESTRICTED
You may specify the account number in FT requests which work with the admission profile. The admission profile does not restrict the account with regard to follow-up processing.

ACCOUNT = <alphanum-name 1..8>
Follow-up processing is to be settled under this account number.

PASSWORD =
With PASSWORD, you specify, where applicable, the BS2000 password for the user ID under which the follow-up processing is to be executed. Here, you can enter a PASSWORD when the user ID in question doesn't have such a password (yet).

PASSWORD = *SAME
The value *SAME is only valid if the PROCESSING-ADMISSION refers to your own user ID. If PASSWORD=*OWN is entered on USER-ADMISSION, then the BS2000 password valid at the time of the request is used for the PROCESSING-ADMISSION. The entry *SAME is only possible here if the follow-up processing is not started with the /ENTER command.

PASSWORD = *NOT-RESTRICTED
You may specify the password in FT requests which work with the admission profile. The admission profile does not restrict the password with regard to follow-up processing.

PASSWORD = *NONE
FT requests which use this admission profile can only initiate follow-up processing on user IDs without a password.

PASSWORD = <c-string 1..8> / <c-string 9..32> / <x-string 1..16>
FT requests which use this admission profile may only initiate follow-up processing on user IDs which are protected with this BS2000 password.

PASSWORD = *SECRET
The system prompts you to enter the password. The entry does not appear on the screen.

SUCCESS-PROCESSING =
With SUCCESS-PROCESSING, you can restrict the follow-up processing which an FT request is permitted to initiate in your system after a successful data transfer.

SUCCESS-PROCESSING = *NOT-RESTRICTED
In FT requests which use this admission profile the operand SUCCESS-PROCESSING may be used without restriction.

SUCCESS-PROCESSING = *NONE
The admission profile does not permit follow-up processing after successful data transfer.

SUCCESS-PROCESSING = <c-string 1..1000 with-low>
Specifies the BS2000 commands which are executed in the local system after successful data transfer. Individual commands must be preceded by a slash (/). The individual commands must be separated by a semicolon (;). If a character string is enclosed by single or double quotes (’ or ”) within a command sequence, *open*FT (BS2000) does not interpret any semicolons within this character string as a separator.

SUCCESS-PROCESSING = *EXPANSION(...)
If a SUCCESS-PROCESSING was specified in an FT request which uses this admission profile, FTAC adds the prefix or suffix specified here to this command. As follow-up processing, the command which has been thus expanded is then executed.

Example

If PREFIX='/PRINT-FILE␣' is defined and SUCC='filename' specified in the FT request, then FT executes the command "/PRINT-FILE␣filename" as follow-up processing.

If a suffix or prefix is defined at this point, then no command sequence for the follow-up processing may be specified in FT requests which use this admission profile. This makes the setting of prefixes and suffixes mandatory.

PREFIX = *NOT-RESTRICTED
Follow-up processing is not restricted by a prefix.

PREFIX = <c-string 1..999 with-low>
The specified prefix is set in front of a command which is specified in an FT request as follow-up processing. Then, the command which has been expanded with the prefix is executed as follow-up processing.

SUFFIX = *NOT-RESTRICTED
The follow-up processing is not restricted by a suffix.

SUFFIX = <c-string 1..999 with-low>
The specified suffix is added to a command which is specified in an FT request as follow-up processing. Then, the command which has been expanded with the suffix is executed as follow-up processing.

FAILURE-PROCESSING =
With FAILURE-PROCESSING, you can restrict the follow-up processing which an FT request is permitted to initiate in your system after a failed data transfer.

FAILURE-PROCESSING = *NOT-RESTRICTED
In FT requests which use this admission profile the operand FAILURE-PROCESSING may be used without restriction.

FAILURE-PROCESSING = *NONE
The admission profile does not permit follow-up processing after failed data transfer.

FAILURE-PROCESSING = <c-string 1..1000 with-low>
Specifies the BS2000 commands which are executed in the local system after failed data transfer. Individual commands must be preceded by a slash (/). The individual commands must be separated by a semicolon (;). If a character string is enclosed by single or double quotes (' or ") within a command sequence, *open*FT (BS2000) does not interpret any semicolons within this character string as a separator.

FAILURE-PROCESSING = *EXPANSION(...)
If a FAILURE-PROCESSING was specified in an FT request which uses this admission profile, FTAC adds the prefix or suffix specified here to this command. As follow-up processing, the command which has been thus expanded is then executed.

If a suffix or prefix is defined at this point, then no command sequence for the follow-up processing may be specified in FT requests which use this admission profile. This makes the setting of prefixes and suffixes mandatory.

PREFIX = *NOT-RESTRICTED
Follow-up processing is not restricted by a prefix.

PREFIX = <c-string 1..999 with-low>
The specified prefix is set in front of a command which is specified in an FT request as follow-up processing. Then, the command which has been expanded with the prefix is executed as follow-up processing.

SUFFIX = *NOT-RESTRICTED
The follow-up processing is not restricted by a suffix.

SUFFIX = <c-string 1..999 with-low>
The specified suffix is added to a command which is specified in an FT request as follow-up processing. Then, the command which has been expanded with the suffix is executed as follow-up processing.

WRITE-MODE =
With WRITE-MODE, you determine the WRITE-MODE specification which is valid for this FT request. WRITE-MODE is only effective if the receive file is in the same system as the admission profile definition.

WRITE-MODE = *NOT-RESTRICTED
In an FT request which accesses this admission profile, the operand WRITE-MODE may be used without restrictions.

WRITE-MODE = *NEW-FILE
In the FT request, *NEW-FILE, *REPLACE-FILE or *EXTEND-FILE may be entered for WRITE-MODE. If the receive file already exists, the transfer will be rejected.

WRITE-MODE = *REPLACE-FILE
In the FT request, only *REPLACE-FILE or *EXTEND-FILE may be entered for WRITE-MODE.

WRITE-MODE = *EXTEND-FILE
In the FT request, only *EXTEND-FILE may be entered for WRITE-MODE.

FT-FUNCTION =
This operand permits the restriction of the profile validity to certain FT functions (=file transfer and file management functions).

FT-FUNCTION = *NOT-RESTRICTED
The full scope of FT functions is available. For reasons of compatibility, the specification NOT-RESTRICTED means that FILE-PROCESSING is not permitted! All other functions are permitted if this value is specified.

FT-FUNCTION = (*TRANSFER-FILE, *MODIFY-FILE-ATTRIBUTES, *READ-DIRECTORY, *FILE-PROCESSING)
The following file transfer functions are available:

***TRANSFER-FILE**
The admission profile may be used for the file transfer functions “transfer files”, “view file attributes” and “delete files”.

***MODIFY-FILE-ATTRIBUTES**
The admission profile may be used for the file transfer functions “view file attributes” and “modify file attributes”.

***READ-DIRECTORY**
The admission profile may be used for the file transfer functions “view directories” and “view file attributes”.

***FILE-PROCESSING**
The admission profile may be used for the “preprocessing” and “postprocessing” file transfer function. The “transfer files” function must also be permitted.

The *FILE-PROCESSING specification is of relevance only for FTAC profiles without a filename prefix. Otherwise the first character of the filename prefix determines whether only normal data transfer (no pipe symbol |) or only preprocessing and postprocessing (pipe symbol |) are to be possible with this FTAC profile.

USER-INFORMATION =
The user and the FTAC administrator can enter a text in the admission profile. This text is displayed with the command SHOW-FT-PROFILE.

USER-INFORMATION = *NONE
No text is stored in the profile.

USER-INFORMATION = <c-string 1..100 with-low>
Here, you enter a character string containing user information.

DATA-ENCRYPTION =
Using DATA-ENCRYPTION, you can restrict the encryption option for user data.

DATA-ENCRYPTION = *NOT-RESTRICTED
The encryption option for user data is not restricted. Both encrypted and unencrypted file transfers are accepted.

DATA-ENCRYPTION = *NO
Only those file transfers which do not have encrypted user data are accepted, i.e. encrypted requests are rejected.
If the request is made in a BS2000, for example, it must be specified there in the NCOPY request DATA-ENCRYPTION=*NO.

DATA-ENCRYPTION = *YES
Only those file transfer requests that have encrypted user data are accepted, i.e.. unencrypted requests are rejected.
If the request is made in a BS2000, for example, it must be specified there in the NCOPY request DATA-ENCRYPTION=*YES.

When using restrictions for FILE-NAME, SUCCESS-PROCESSING and FAILURE-PROCESSING, keep in mind that
- a restriction for follow-up processing must always be made for SUCCESS- and FAILURE-PROCESSING. Otherwise, it is possible that users will avoid this step.
- PREFIX of FILE-NAME, SUCCESS-PROCESSING and FAILURE-PROCESSING must correspond
  (e.g. FILE-NAME=*EXP(XYZ.),SUCC=*EXP('/PRINT-FILE XYZ.').

Example 1

Dagobert Duck wishes to create an admission profile for the following purpose:

Dussel Duck, employee at the Duck Goldmine, has his own BS2000 computer. He has to transfer monthly reports on a regular basis to his boss Dagobert‘s computer, DAGODUCK, using File Transfer. The file needs to have the name MONTHLYREPORT.GOLDMINE and is to be printed out after transfer. Since Dagobert’s admission set does not permit any “inbound” requests, he needs to give the profile privileged status (he is permitted to do this, since he is an FTAC administrator). The Goldmine computer has the security level 50. The command required to create such an admission profile is as follows:

```
/CREATE-FT-PROFILE NAME=GOLDMORE,                                    -
/                    TRANSFER-ADMISSION='monthlyreportfortheboss',   -
/                    PRIVILEGED=*YES,                                -
/                    IGNORE-MAX-LEVELS=(INBOUND-RECEIVE=*YES,        -
/                    INBOUND-PROCESSING=*YES),                       -
/                    TRANSFER-DIRECTION=*FROM-PARTNER,               -
/                    PARTNER=GOLDMINE,                               -
/                    FILE-NAME=MONTHLYREPORT.GOLDMINE,               -
/                    SUCCESS-PROCESSING=                             -
/                    '/PRINT-FILE␣MONTHLYREPORT.GOLDMINE',           -
/                    FAILURE-PROCESSING=*NONE,                       -
/                    WRITE-MODE=*REPLACE-FILE
```

The short form of this command is:

```
/CRE-FT-PROF␣GOLDMORE,TRANS-AD='monthlyreportfortheboss',            -
/PRIV=*YES,IGN-MAX-LEV=(I-R=*YES,I-P=*YES),TRANS-DIR=*FROM           -
/PART=GOLDMINE, FILE-NAME=MONTHLYREPORT.GOLDMINE,                    -
/SUCC='/PRINT-FILE␣MONTHLYREPORT.GOLDMINE',FAIL=*NONE,               -
/WRITE=*REPL
```

File management can also be performed with this admission profile (see the specifications for the IGNORE-MAX-LEVELS operand).

Dussel Duck, who keeps the monthly report for the goldmine in his BS2000 computer in the file NOTHINGBUTLIES, can use the following *open*FT for BS2000 command to send it to the central computer DAGODUCK and print it out there:

```
/TRANSFER-FILE␣TO,DAGODUCK,(NOTHINGBUTLIES),
                (FILE=*NOT-SPECIFIED,TRANS-AD='monthlyreportfortheboss')
```

If you only want to use FTAC profiles for the ftexec command then you must specify a filename or filename prefix that starts with the character string '|ftexecsv'.

If the entire command string is predefined as the filename in the profile then only an empty string (‘ ‘) may be passed as the filename when ftexec is called (at a PC/UNIX system). You must make sure that the -t and/or -b parameters in the profile and ftexec call match, for example:

Profile with FILE-NAME=‘|ftexecsv “/SHOW-FT-PARTNER HUGO” -t’

```
Call: ftexec -t <hostname> "" <transfer-admission>
```

There are two ways of specifying a command prefix for ftexec:

```
FILE-NAME=*EXP('|ftexecsv <command-prefix>')
FILE-NAME=*EXP('|ftexecsv -p=<command-prefix>')
```

In the first form, the command string used to call ftexec is not subject to any restrictions, whereas a command sequence is rejected in the second form.

If the command sequence or command prefix set in the profile for the ftexec call contain spaces then these must be quoted (‘ ‘). Any quote marks that appear in the command sequence must themselves be quoted.

Example 2

You want to create a profile which can be used to run precisely one file processing command. A number of logging records are output in the example below.

```
/CR-FT-PRO NUR1VORV,,'GetLoggingRecords'            -
,FILE-NAME=*EXP('|ftexecsv -p="/SH-FT-LOG-REC ,"')  -
,FT-FUN=(*TRANS-F,*FILE-PROC)
```

The following command, for example, can be used to access the profile from UNIX system or Windows system:

```
ftexec FTBS2 3 GetLoggingRecords
```

The last three logging records to be written in the BS2000 system are output:

```
/sh-ft-log-rec ,3
%TYP LOGG-ID TIME    RC    PARTNER  INITIATOR INIT USER-ADM FILENAME
%2000-11-20
%C   275800 15:59:36 0000 >UNIX    *REMOTE        UID     |FTEXECSV -P="/SH-FT-LOG-REC ,"
3 -B
%T   275799 15:55:55 0000 <FTBS2    UID       1L3I UID      :CAT2:$UID.CH.1
%T   275798 15:55:54 0000 >FTBS2    *REMOTE        UID       2
```

Command return codes

(SC2)	SC1	Maincode	Meaning
0	0	FTC0051	A user ID with the same name already exists in the system.
0	0	FTC0056	Transfer admission is blocked.
0	64	FTC0100	An FT profile with the same name already exists.
0	64	FTC0101	An FT profile with the same Transfer-Admission already exists.
0	64	FTC0150	The access password is missing.
0	64	FTC0153	The owner identification entered is not the own user ID.
0	64	FTC0157	No authorization to create the profile.
0	64	FTC0172	The User-Admission entered does not exist in the system.
0	64	FTC0173	The Processing-Admission entered does not exist in the system.
0	64	FTC0178	The partner name entered occurs several times.
0	64	FTC0182	Maximum length for partner names has been exceeded.
0	64	FTC0200	The total length of the two follow-up processing commands is too long.
0	64	FTC0255	A system error has occurred.

SC1/2 = Subcode 1/2 in decimal notation

For additional information, see section “Command return codes” on page 102

4.12 DELETE-FT-INSTANCE
Delete the administration entry of an *open*FT instance

This command deletes the administration entry of the instance. All of the variable data such as, for example, the request file are kept and can be re-activated with the same instance name by re-executing the CREATE-FT-INSTANCE command.

In the event that a user task has altered the deleted instance, this will only be recognized on the next attempt by *open*FT to access this instance. *open*FT commands for this instance are rejected in this case, issuing the message FTR1025. The user must set another instance using the SET-FT-INSTANCE command.

DELETE-**FT**-**INST**ANCE
NAME = <alphanum-name 1...8>

Operand

NAME = <alphanum-name 1..8>
The name of the *open*FT instance that is to be deleted. The default instance cannot be deleted.

Command return codes

(SC2)	SC1	Maincode	Meaning
83	32	CMD0221	Internal error.
24	64	FTR1024	Standard instance must not be deleted.
25	64	FTR1025	Instance does not exist.

SC1/2 = Subcode 1/2 in decimal notation
For additional information, see section “Command return codes” on page 102

4.13 DELETE-FT-KEY-SET (FTDELKEY) Delete a key pair set

Using this command, you are deleting the key pair set of a reference. The key pair consists of a private key, which is internally administered by *open*FT, and a public key. Public keys are stored on the configuration user ID of the *open*FT instance (default: $SYSFJAM) under the name **SYSPKF.R<key reference>.L<key length>**. The key reference is a numeric designator for the version of the key pair. For each reference there are two keys with lengths of 768 and 1024 respectively.

A key pair set should only be deleted if no partner system uses the corresponding public key any longer. This means that, after creating a new key pair set using CREATE-FT-KEY-SET, the new public key should be made available to all of the partner systems in which the local system is to be authenticated.

There should always be at least one key pair set in your *open*FT instance, otherwise all requests will be carried out in unencrypted form.

DELETE**-FT-KEY**-SET / **FTDELKEY**
REFERENCE = <integer 1..9999999>

Operands

REFERENCE = <integer 1..9999999>
Allows selection of the key pair set to be deleted. You will find the reference in the name of the public key file (see above).

Command return codes

(SC2)	SC1	Maincode	Meaning
83	32	CMD0221	Internal error
2	0	FTR1030	Warning: last key pair deleted
32	64	FTR1032	Last key pair must not be deleted
35	64	FTR1035	User not authorized for this command.
37	64	FTR1037	Key reference unknown

SC1/2 = Subcode 1/2 in decimal notation
For additional information, see section “Command return codes” on page 102

Example

Delete the key pair set with the public keys $SYSFJAM.SYSPKF.R137.L768 and $SYSFJAM.SYSPKF.R137.L1024:

```
/DELETE-FT-KEY-SET REF=137
```

4.14 DELETE FT-LOGGING-RECORDS (FTDELLOG) Delete logging records

With DELETE-FT-LOGGING-RECORDS (or FTDELLOG) you can delete FT logging records for all login names. This function is not permitted for the ordinary user.

In principle, *open*FT can write any number of logging records (until the disk is full). The FT administrator should save the existing logging records (e.g. in hardcopy, to tape or as a file in CSV format) and at regular intervals (weekly, for example, if there is a large number of requests) and delete older logging records. This means, firstly, that logging records are retained for a long period, thereby ensuring continuous documentation, and secondly, that memory space is not occupied unnecessarily.

The logging records are saved by redirecting the output of SHOW-FT-LOGGING-RECORDS (Displaying logging records, page 249ff) to a file in CSV format (for more information, see SHOW-FT-LOGGING-RECORDS) or to a printer e.g.

```
/ASSIGN-SYSLST LOGGING-DATEI
/SHOW-FT-LOGGING-RECORDS...,NUMBER=*ALL,OUTPUT=*SYSLST(*CSV)
```

When backing up logging records, CSV format should be preferred to the default format since in this format all the information is backed up “in a single line” and a variety of tools can be used for the further processing of the information.

When deleting logging records, the PAM pages occupied by the log file (SYSLOG on the configuration user ID of the *open*FT instance, (default: $SYSFJAM) are not released; the free space within the file is, however, used to store new records.

When FTAC functionality is used, FTAC also writes logging records. These logging records can only be deleted by the FTAC administrator using DELETE-FT-LOGGING-RECORDS.

Command execution may take several minutes, depending on the size of the log file. To prevent inconsistencies, it is not possible to use the K2 key to interrupt the command.

DELETE-**FT-LOG**GING-**REC**ORDS **/ FTDELLOG**

SELECT = ***ALL** / ***OWN** / ***PAR**AMETERS(...)

***PAR**AMETERS(...)

OWNER-**ID**ENTIFICATION = ***ALL** / ***OWN** / <name 1..8>

,**LOGG**ING-**DATE** = ***TODAY** / ***TOMOR**ROW / <date 8..10>

,**LOGG**ING-**TIME** = **00:00** / <time 1..8>

,**REC**ORD-**TYPE** = ***ALL** / ***PAR**AMETERS(...)

***PAR**AMETERS(...)

FT = ***ALL** / ***NONE**

,**FTAC** = ***ALL** / ***NONE**

,**LOGG**ING-**ID** = ***ALL** / <integer 1..99999999>

Operands

SELECT =
Used to select a group of logging records.

SELECT = *ALL
Deletes all logging records.

SELECT = *OWN
Deletes all logging records of your own ID.

SELECT = *PARAMETERS(...)

OWNER-IDENTIFICATION =
User ID whose logging records are to be deleted.

OWNER-IDENTIFICATION = *ALL
The user ID is not a selection criterion.

OWNER-IDENTIFICATION = *OWN
Logging records in the user ID are deleted.

OWNER-IDENTIFICATION = <name 1..8>
User ID whose logging records are to be deleted.

LOGGING-DATE =
Date before which the logging records are to be deleted.

LOGGING-DATE = *TODAY
If a time was specified explicitly with LOGGING-TIME, all logging records that were written before this time are deleted. If no date was specified, all logging records are deleted that were written up to midnight inclusive of the previous day.

LOGGING-DATE = *TOMORROW
All logging records that were created before the command was input are deleted.

LOGGING-DATE = <date 8..10>
Date in the format *yyyy-mm-dd* or *yy-mm-dd*, e.g. 2006-12-24 or 06-12-24 for the 24th of December, 2006. FT then deletes only those logging records that were written before the date and time specified with LOGGING-TIME and LOGGING-DATE.

LOGGING-TIME =
Logging records written up to the specified time are deleted.

LOGGING-TIME = 00:00
If a date was specified explicitly with LOGGING-DATE, FT deletes all logging records written before the specified date. If no date was specified, FT deletes all logging records that were written up to midnight inclusive of the previous day.

LOGGING-TIME = <time 1..8>
Time for the day specified with LOGGING-DATE. FT deletes all logging records written before this time. Your entry must have the format *hh:mm:ss*, e.g. 14:30:10.

RECORD-TYPE =
Defines the type of logging records to be deleted.

RECORD-TYPE = *ALL
The record type is not a selection criterion.

RECORD-TYPE = *PARAMETERS(...)
Type of the logging record.

FT = *ALL / *NONE
Specifies whether or not the FT logging records are to be deleted.

FTAC = *ALL / *NONE
Specifies whether or not FTAC logging records are to be deleted.
Please note that the FTAC logging records can only be deleted by the FTAC administrator.

LOGGING-ID =
Used to select the logging records on the basis of the logging ID.

LOGGING-ID = *ALL
The logging ID is not a selection criterion.

LOGGING-ID = <integer 1..99999999>
All logging records with a logging ID smaller than or equal to the specified value are deleted.

Command return codes

(SC2)	SC1	Maincode	Meaning
0	0	CMD0001	No log records available for the selection criteria.
83	32	CMD0221	Internal error
34	64	FTR1034	Command only permissible for FT or FTAC administrator.
35	64	FTR1035	User not authorized for this command.
36	64	FTR1036	User not authorized for other user Ids.

SC1/2 = Subcode 1/2 in decimal notation
For additional information, see section “Command return codes” on page 102

Example

The FT administrator wishes to delete all FT logging records.

```
/DELETE-FT-LOGGING-RECORDS  SELECT=*PARAMETERS(LOGGING-DATE=*TOMORROW, -
/                           RECORD-TYPE=*PARAMETERS(FTAC=*NONE))
```

4.15 DELETE-FT-PROFILE Delete admission profile

A prerequisite for using this command is the use of *open*FT-AC.

With the command DELETE-FT-PROFILE, you can delete admission profiles. You should go through your admission profiles on a regular basis and delete all those which are no longer required, so that they don't take up memory.
With SHOW-FT-PROFILE (see page 279ff), you can view the profiles and decide which ones you no longer need.

The FTAC administrator can delete the admission profiles of any FTAC user using DELETE-FT-PROFILE. Of course, he should inform them of his intentions ahead of time.

```
DELETE-FT-PROFILE

NAME = *ALL / <alphanum-name 1..8>
,PASSWORD = *NONE / <c-string 1..8 with-low> / <x-string 1..16> / *SECRET
,SELECT-PARAMETER = *OWN / *PARAMETERS(...)
    *PARAMETERS(...)
        TRANSFER-ADMISSION = *ALL / *NOT-SPECIFIED / <alphanum-name8..32> /
                             <c-string 8..32 with-low> / <x-string 15..64> / *SECRET
        ,OWNER-IDENTIFICATION = *OWN / *ALL / <name 1..8>
```

Operands

NAME =
With NAME, you can access the admission profile to be deleted using its name.

NAME = *ALL
With *ALL, you can delete all admission profiles. The FTAC user can delete all of his admission profiles with this operands, if he does not select a special profile with SELECT-PARAMETER.
The administrator can delete his own profiles with this entry. He can also use SELECT-PARAMETER to delete all the admission profiles of a particular user or all the admission profiles in the system.

NAME = <alphanum-name 1..8>
You wish to delete the admission profile with the specified name.

PASSWORD =
With PASSWORD, you enter the FTAC password which permits you to use FTAC commands with your user ID.

PASSWORD = *NONE
No FTAC password is required.

PASSWORD = <c-string 1..8 with-low> / <x-string 1..16>
Specifies the corresponding FTAC password.
If the FTAC administrator has defined an FTAC password, then this password must be entered here, if he wishes to delete the profiles of other users.

PASSWORD = *SECRET
The system prompts you to enter the password. However, the password does not appear on the screen.

SELECT-PARAMETER =
With SELECT-PARAMETER, you can enter selection criteria for the admission profiles to be deleted. The following options are available:

– FTAC users can address the admission profiles to be deleted using their TRANSFER ADMSSION.

– FTAC administrators can address the admission profiles to be deleted using their TRANSFER ADMSSION or OWNER IDENTIFICATION.

SELECT-PARAMETER = *OWN
With *OWN, you delete your own admission profiles.

SELECT-PARAMETER = *PARAMETERS(...)
With this structure, you can enter individual selection criteria.

TRANSFER-ADMISSION =
With TRANSFER-ADMISSION, you can use the transfer admission of an admission profile as a selection criterion for deletion.

TRANSFER-ADMISSION = *ALL
You wish to delete admission profiles irrespective of the TRANSFER-ADMISSION.

TRANSFER-ADMISSION = *NOT-SPECIFIED
You wish to delete admission profiles for which no transfer admission is specified.

TRANSFER-ADMISSION = <alphanum-name 8..32> /
<c-string 8..32 with-low> / <x-string 15..64>
You wish to delete the admission profile which is accessed with this transfer admission. The alphanumeric entry is always saved in lower-case letters. The FTAC user can only enter the transfer admissions of his own admission profiles.

TRANSFER-ADMISSION = *SECRET
The system prompts you to enter the transfer admission. This does not appear on the screen.

OWNER-IDENTIFICATION =
OWNER-IDENTIFICATION allows you to delete a specific owner's admission profile. The FTAC user can only delete his own profiles, whereas the FTAC administrator can also enter foreign user IDs.

OWNER-IDENTIFICATION = *OWN
You wish to delete your own admission profile.

OWNER-IDENTIFICATION = *ALL
This entry allows the FTAC administrator to delete admission profiles of all user IDs. The FTAC user is not permitted to use this entry.

OWNER-IDENTIFICATION = <alphanum-name 1..8>
You wish to delete the admission profiles under this user ID.

Command return codes

(SC2)	SC1	Maincode	Meaning
0	64	FTC0053	No FT profile exists with these criteria.
0	64	FTC0150	The access password is missing.
0	64	FTC0153	The owner identification entered is not the user's own ID.
0	64	FTC0255	A system error occurred.

SC1/2 = Subcode 1/2 in decimal notation
For additional information, see section "Command return codes" on page 102

4.16 EXPORT-FTAC-ENVIRONMENT Export FTAC admission profiles and sets

*open*FT-AC must be installed to use this command.

The FTAC administrator can easily “move” admission profiles and sets when a user migrates from one computer to another. The commands EXPORT-FTAC-ENVIRONMENT and IMPORT-FTAC-ENVIRONMENT are intended for this purpose.

This command is not available to FTAC users! Export files cannot be extended. They must be deleted and created again if necessary.

The commands only affect the currently set *open*FT instance. If necessary, the FTAC administrator must create them under several *open*FT instances.

```
EXPORT-FTAC-ENVIRONMENT

TO-FILE = <filename 1..54>
,USER-IDENTIFICATION = *ALL / list-poss(100): <name 1..8>
,SELECT-PARAMETER = *ALL / *PARAMETERS(...)
    *PARAMETERS(...)
        PROFILE-NAME = *ALL / *NONE / list-poss(100): <alphanum-name 1..8>
        ,ADMISSION-SET = *YES / *NO
```

Operands

TO-FILE = <filename 1..54>
Name of the file in which the admission profiles and sets are output. Temporary files may not be used.

USER-IDENTIFICATION =
The user ID whose admission profiles and sets are to be output on file.

USER-IDENTIFICATION = *ALL
The admission profiles and sets of all user IDs are to be output on file.

USER-IDENTIFICATION = list-poss(100): <name 1..8>
The admission profiles and sets of the user IDs specified are to be output on file.

SELECT-PARAMETER =
This determines whether only admission profiles, only admission sets, or both are to be output on file. For admission profiles, you can select those which are to be output.

SELECT-PARAMETER = *ALL
All admission profiles and sets associated with the user ID specified under USER-IDENTIFICATION are to be output on file.

SELECT-PARAMETER = *PARAMETERS(...)
This is used to specify which of the admission profiles and sets associated with the USER-IDENTIFICATION are to be output on file.

PROFILE-NAME = *ALL
All admission profiles are output on file.

PROFILE-NAME = *NONE
No admission profiles are exported.

PROFILE-NAME = list-poss(100): <alphanum-name 1..8>
Only the profiles with the specified names (maximum 100) are output on file.

ADMISSION-SET = *YES
All admission sets are output on file.

ADMISSION-SET = *NO
No admission sets are exported.

Command return codes

(SC2)	SC1	Maincode	Meaning
0	0	FTC0054	No information matches the specified criteria.
0	64	FTC0102	File already exists.
0	64	FTC0104	Access to the user ID denied or the user ID does not exist.
0	64	FTC0105	Access to the file denied.
0	64	FTC0106	Access to the temporary file denied.
0	64	FTC0156	The command may only be executed by the FTAC administrator.
0	64	FTC0180	The USER-ID entered occurs several times.
0	64	FTC0181	The FT profile name entered occurs several time.
0	64	FTC0206	The export file cannot be created.
0	64	FTC0255	A system error occurred.

SC1/2 = Subcode 1/2 in decimal notation
For additional information, see section “Command return codes” on page 102

4.17 IMPORT-FTAC-ENVIRONMENT Import FTAC admission profiles and sets

*open*FT-AC must be installed to use this command.

The FTAC administrator can easily “move” admission profiles and sets when a user migrates from one computer to another. The commands EXPORT-FTAC-ENVIRONMENT and IMPORT-FTAC-ENVIRONMENT are intended for this purpose. These commands cannot be used by the FTAC user.

If the FTAC administrator does not possess TSOS privileges then all imported admission profiles will be locked. This can be seen in the SHOW-FT-PROFILE command in the specification *LOCKED (by_import). Privileged profiles lose their privileged status when imported. They will also be designated as private.

These restrictions are not valid by default, if the FTAC administrator also has the TSOS privilege. In this case, profiles are imported unlocked and privileges are retained. If that is not desirable due to security concerns, the FTAC administrator can force locking by specifying the SECURITY=HIGH parameter.

An admissions profile is otherwise only imported if its name does not exist on the destination ID.

If the target computer already has an admission profile with the same transfer admission and the admission profile is designated as private, both transfer admissions are locked. The transfer admission of the old profile is set to *DUPLICATED and the transfer admission of the imported profile is set to *NOT-SPECIFIED. If the already existing admission profile is designated as ”public”, then it is not locked.

IMPORT-**FTAC-ENV**IRONMENT

FROM-FILE = <filename 1..54>

,**USER-ID**ENTIFICATION = ***ALL** / list-poss(100): <name 1..8>

,**SEL**ECT-PARAMETER = ***ALL** / ***PAR**AMETERS(...)

***PAR**AMETERS(...)

PROFILE-**NAME** = ***ALL** / ***NONE** / list-poss(100): <alphanum-name 1..8>

,**AD**MISSION-**SET** = ***YES** / ***NO**

,**SECURITY** = ***STD** / ***HIGH**

Operands

FROM-FILE = <filename 1..54>
Name of the file from which the admission profiles and sets are to be imported. Temporary files may not be used. If the file contains invalid data or if there is an error while accessing the file, the command is rejected with the message FTC0103.

USER-IDENTIFICATION =
This is the user ID whose admission profiles and sets are to be transferred from an export file.

USER-IDENTIFICATION = *ALL
The admission profiles and sets of all users are to be transferred.

USER-IDENTIFICATION = list-poss(100): <name 1..8>
The admission profiles and sets of the users specified (maximum 100) are to be transferred.

SELECT-PARAMETER =
This determines whether only admission profiles, only admission sets, or both are to be imported. For admission profiles, you can specify which are to be imported.

SELECT-PARAMETER = *ALL
All the admission profiles and sets associated with the user ID specified under USER-IDENTIFICATION are to be imported.

SELECT-PARAMETER = *PARAMETERS(...)
Here, you can specify which of the admission profiles and sets associated with the USER-IDENTIFICATION are to be imported.

PROFILE-NAME = *ALL
All admission profiles are to be imported.

PROFILE-NAME = *NONE
No admission profiles are to be imported.

PROFILE-NAME = list-poss(100): <alphanum-name 1..8>
Only the profiles specified are to be imported (maximum 100).

ADMISSION-SET = *YES
All admission sets are to be imported.

ADMISSION-SET = *NO
No admission sets are to be imported.

SECURITY =
An FTAC administrator with TSOS can use this operand to control security.

SECURITY = *STD
For FTAC administrators with TSOS privilege:
The profile attributes are not altered when imported.

For FTAC administrators not having the TSOS privilege:
This operand works like the specification *HIGH, i.e. the admissions profiles are locked (locked by import) and retain the attributes USAGE=PRIVATE and PRIVILEGED = NO.

SECURITY = *HIGH
The admissions profiles are locked (locked by import) and retain the attributes USAGE=PRIVATE and PRIVILEGED=NO.

Command return codes

(SC2)	SC1	Maincode	Meaning
0	64	FTC0052	The information output was interrupted.
0	0	FTC0054	No information matches the specified criteria
0	0	FTC0056	The transfer admission is locked.
0	64	FTC0100	An FT profile with the specified name already exists.
0	64	FTC0101	An FT profile with the specified transfer admission already exists.
0	64	FTC0103	The file is not an FTAC export file or access is denied.
0	64	FTC0104	Access to the user ID denied or the ID does not exist.
0	64	FTC0105	Access to the file denied.
0	64	FTC0106	Access to the temporary file denied.
0	64	FTC0156	The command can only be executed by the FTAC administrator.
0	64	FTC0177	The filename entered is unknown.
0	64	FTC0180	The USER-ID entered occurs several times.
0	64	FTC0181	The FT profile name entered occurs several times.
0	64	FTC0255	A system error occurred.

SC1/2 = Subcode 1/2 in decimal notation
For additional information, see section “Command return codes” on page 102

4.18 MODIFY-FILE-TRANSFER (FTMODREQ) Modify request queue

You use the MODIFY-FILE-TRANSFER command (or FTMODREQ) to modify the position and priority of your outbound requests within the *open*FT request queue. You have the option of processing the outbound requests in any order you wish. Newly input requests or requests whose priority changes are put at the end of the request queue for the corresponding priority. If already active requests are repositioned behind waiting outbound requests, the active requests are interrupted if possible in favor of those waiting.

MODIFY-FILE-TRANSFER is only valid for outbound requests.

MODIFY-**FILE-T**RANSFER / **FTMODREQ**

TRANSFER-ID = ***ALL** / <integer 1..2147483639>
,**SEL**ECT = ***OWN** / ***PAR**AMETERS(...)
***PAR**AMETERS(...)
OWNER-**ID**ENTIFICATION = ***OWN** / ***ALL** / <name 1..8>
,**PART**NER = ***ALL** / <text 1..200 with-low>
,**FILE** = ***ALL** / <filename 1..54> / <c-string 1..512 with-low> / ***LIB**RARY-ELEMENT(...)
***LIB**RARY-ELEMENT(...)
LIBRARY = ***ALL** / <filename 1..54>
,**ELEM**ENT = ***ALL** / <filename 1..64 without-gen-vers>(...) /
<composed-name 1..64 with-under>(...)
<filename>(...) / composed-name 1..64>(...)
VERSION = ***ALL** / <text 1..24>
,**TYPE** = ***ALL** / <name 1..8>
,**MONJV** = ***NONE** / <filename 1..54>
,**JV-PASS**WORD = ***NONE** / <c-string 1..4> / <x-string 1..8> /
<integer -2147483648..2147483647> / ***SECRET**
,**QUEUE-POS**ITION = ***UNCHANGED** / ***FIRST** / ***LAST**
,**PRIO**RITY = ***UNCHANGED** / ***NORMAL** / ***HIGH** / ***LOW**

Operands

TRANSFER-ID =
Transfer ID of the outbound request to be modified.

TRANSFER-ID = *ALL
Modifies all outbound requests, If further selections haven't been specified with SELECT (see below).

TRANSFER-ID = <integer 1..2147483639>
Transfer ID which is communicated to the local system in the FT request confirmation.

SELECT =
Contains selection criteria for outbound requests to be modified. A request is only modified if all the criteria specified are met. This means that the criteria must be specified accordingly in the FT request.

SELECT = *OWN
Modifies all FT requests of the user's own ID with the TRANSFER-ID specified.

SELECT = *PARAMETERS(...)

OWNER-IDENTIFICATION =
Identifies the owner of the FT request. Users may only enter their own user ID.

OWNER-IDENTIFICATION = *OWN
Modifies only outbound requests with the user's own ID.

OWNER-IDENTIFICATION = *ALL
Modifies outbound requests for all user IDs. Only the FTAC administrator may use this entry.

OWNER-IDENTIFICATION = <name 1..8>
Specifies a user ID whose requests are to be modified. Users may only enter their own user ID.

PARTNER =
Modifies outbound requests which are to be executed with a particular partner system.

PARTNER = *ALL
The name of the partner system is not selected as a criterion for the outbound requests to be modified.

PARTNER = <text 1..200 with-low>
Outbound requests are modified which are to be executed with this partner system. You can specify the name from the partner list or the address of the partner system. For more information on address specifications, see section "Specifying partner addresses" on page 39.

FILE =
Modifies outbound requests which access this file or library member in the local system as a send or receive file. The file or library member name must be entered exactly as in the file transfer request.

FILE = *ALL
The filename is not selected as a criterion for the outbound requests to be modified.

FILE = <filename 1..54> / <c-string 1..512 with-low>
Modifies outbound requests which access this file (DVS/POSIX) in the local system.

FILE = *LIBRARY-ELEMENT(...)
Modifies outbound requests which access library members in the local system.

LIBRARY =
Selects the library.

LIBRARY = *ALL
The library name is not selected as a criterion for the outbound requests to be modified.

LIBRARY = <filename 1..54>
Outbound requests are to be modified which access this library.

ELEMENT =
Selects the library member.

ELEMENT = *ALL
The name of the library member is not selected as a criterion for the outbound requests to be modified.

ELEMENT = <filename 1..64 without-gen-vers>(...)/
<composed-name 1..64 with-under>(...)
Name of the library member.

VERSION =
Version of the member.

VERSION = *ALL
The library member version is not selected as a criterion for the outbound requests to be modified.

VERSION = <text 1..24>
Only outbound requests which access this version of the library member are to be modified.

TYPE =
Type of library member.

TYPE = *ALL
The member type is not selected as a criterion for the outbound requests to be modified.

TYPE = <name 1..8>
Only outbound requests which access library members of this type are to be modified.

MONJV =
Selects any outbound request which is monitored by this job variable.

MONJV = *NONE
No job variable is selected as criterion for the interruption of the file transfer.

MONJV = <filename 1..54>
The outbound request monitored by this job variable is to be modified.

JV-PASSWORD =
Specifies the password which is needed to access the job variable.
If you have already entered the password using the BS2000 command ADD-PASSWORD, you do not need to enter JV-PASSWORD.

JV-PASSWORD = *NONE
The job variable is not password-protected.

JV-PASSWORD =
<c-string 1..4> / <x-string 1..8> / <integer -2147483648..2147483647>
This password is required for the job variable.

JV-PASSWORD = *SECRET
The system prompts you to enter the password. The entry does not appear on the screen. However, the password does not appear on the screen.

QUEUE-POSITION =
Defines the new position of the outbound request that is to be modified in the *open*FT request queue. The position of an FTAM request can only be changed relative to the requests that affect the same FTAM partner.

QUEUE-POSITION = *UNCHANGED
The position of the outbound request in this user's *open*FT request queue remains unchanged.

QUEUE-POSITION = *FIRST
The outbound request is placed in front of all the other requests of the same priority issued by the user in the *open*FT request queue

QUEUE-POSITION = *LAST
The outbound request is placed behind all the other requests of the same priority issued by the user in the *open*FT request queue.

PRIORITY =
Modifies the priority of the FT request.

PRIORITY = *UNCHANGED
The priority of the FT request remains unchanged.

PRIORITY = *NORMAL
The priority of the FT request is set to the normal value.

PRIORITY = *HIGH
The FT request is given a high priority.

PRIORITY = *LOW
The FT request is given a low priority.

Command return codes

(SC2)	SC1	Maincode	Meaning
0	0	CMD0001	No requests available for the selection criteria.
82	32	CMD0221	Internal error. Job variable not accessible
83	32	CMD0221	Internal error.
36	64	FTR1036	User not authorized for other user Ids.
47	64	FTR1047	The request with the specified transfer ID could not be found.

SC1/2 = Subcode 1/2 in decimal notation
For additional information, see section “Command return codes” on page 102

Example:

```
/SHOW-FILE-TRANSFER
% TRANS-ID INI   STATE  PARTNER  DIR   BYTE-COUNT FILE-NAME
% 54483612  LOC   WAIT  UNIX1    FROM  0          FILE1
% 11164324  LOC   WAIT  UNIX2    FROM  0          FILE2
%  FTR0004 FT-BS2000: INFORMATION COMPLETED

/MODIFY-FILE-TRANSFER SELECT=(FILE2),QUEUE-POS=FIRST

/SHOW-FILE-TRANSFER
% TRANS-ID  INI  STATE  PARTNER  DIR   BYTE-COUNT FILE-NAME
% 11164324  LOC   WAIT  UNIX2    FROM  0          FILE2
% 54483612  LOC   WAIT  UNIX1    FROM  0          FILE1
```

4.19 MODIFY-FT-ADMISSION-SET Modify admission set

Prerequisite for using this command is the use of *open*FT-AC.

The command MODIFY-FT-ADMISSION-SET can be used to modify the admission set for your own or other user IDs. You may access two components of the admission set:

a) You can define a password to be entered for almost all subsequent FTAC commands (except the /SHOW... commands). This prevents other users working with your user ID from entering FTAC commands.

It is not possible to have an FTAC password output. If an FTAC user forgets his FTAC password, only the FTAC administrator can delete or modify the password.

WARNING!

If the FTAC administrator should assign and subsequently forget a password, *open*FT-AC must be reinstalled. In this case, all admission profiles and sets are deleted! If SECOS is installed, this can be avoided by appointing a new administrator.

b) FTAC users may modify the maximum value for the security levels that can be accessed from their IDs (the MAX-USER-LEVELS) within the range specified by the FTAC administrator. However, users cannot override the limit values specified by the FTAC administrator (MAX-ADM-LEVELS). They can simply reduce the limit values since, in the case of FT requests, FTAC performs the admission check on the basis of the smallest value in the admission set. The MAX-USER-LEVELS are only effective if they are lower, i.e. more restrictive, than the MAX-ADM-LEVELS.

   FTAC administrators assign a maximum security level for each of the six basic functions. The user ID associated with the admission set can then use this function with all partner systems with this security level or lower. The owner of the admission set may only increase the degree of restriction. The admission check for FT requests is conducted by FTAC on the basis of the smallest (most restrictive) value in the admission set.

In addition, the FTAC administrator can delete an admission set from the admission file by entering the default admission set for the user ID in question (MAX-LEVELS=*STD). This is also possible with user IDs which have already been deleted!

```
MODIFY-FT-ADMISSION-SET

 USER-IDENTIFICATION = *OWN / *STD / <alphanum-name 1...8>
,PASSWORD = *NONE / <c-string 1...8 with-low> / <x-string 1...16> / *SECRET
,SELECT-PARAMETER = *ALL
,NEW-PASSWORD = *OLD / *NONE / <c-string 1...8  with-low> / <x-string 1...16> / *SECRET
,MAX-LEVELS = *UNCHANGED / *STD / <integer 0...100> / *PARAMETERS(...)
   *PARAMETERS(...)
      OUTBOUND-SEND = *UNCHANGED / *STD / <integer 0...100>
      ,OUTBOUND-RECEIVE = *UNCHANGED / *STD / <integer 0...100>
      ,INBOUND-SEND = *UNCHANGED / *STD / <integer 0...100>
      ,INBOUND-RECEIVE = *UNCHANGED / *STD / <integer 0...100>
      ,INBOUND-PROCESSING = *UNCHANGED / *STD / <integer 0...100>
      ,INBOUND-MANAGEMENT = *UNCHANGED / *STD / <integer 0...100>
```

Operands

USER-IDENTIFICATION =
Specifies the user ID whose admission set is to be modified.

USER-IDENTIFICATION = *OWN
The admission set for the user ID which you are currently using is to be modified.

USER-IDENTIFICATION = *STD
The default admission set is to be modified. Only the FTAC administrator can make this entry.

USER-IDENTIFICATION = <alphanum-name 1..8>
The admission set for this user ID is to be modified. The FTAC user can only enter his own user ID here, whereas the FTAC administrator can modify the admission set for any user ID.

PASSWORD =
With PASSWORD, you enter the FTAC password which authorizes you to use FTAC commands, if such a password was defined in your admission set. An FTAC password is set with the operand NEW-PASSWORD.

PASSWORD = *NONE
No FTAC password is required for this admission set.

PASSWORD = <c-string 1..8 with-low> / <x-string 1..16>
This password authorizes this user to use FTAC commands.

PASSWORD = *SECRET
The system prompts you to enter the password. However, the password does not appear on the screen.

SELECT-PARAMETER = *ALL
In later *open*FT-AC versions it will be possible to specify additional selection criteria here.

NEW-PASSWORD =
With this operand, you change the FTAC password. If such an FTAC password has already been set, it must be used for almost all FTAC commands on the user ID for this admission set (except: the SHOW...- commands). This is done using the parameter PASSWORD in the respective commands.

NEW-PASSWORD = *OLD
The FTAC password remains unchanged.

NEW-PASSWORD = *NONE
No FTAC password is required for the user ID associated with this admission set.

NEW-PASSWORD = <c-string 1..8 with-low> / <x-string 1..16>
Specification of the new FTAC password.

NEW-PASSWORD = *SECRET
The system prompts you to input the password. The input does not appear on the screen, however.

MAX-LEVELS =
With this operand, you set which security level(s) you can access, with which basic functions, from the user ID of this admission set. Either you can set one security level for all basic functions or different security levels for each basic function. The MAX-USER-LEVELS for this admission set are subordinate to the settings made by the FTAC administrator (see example for SHOW-FT-ADMISSION-SET, page 244). FTAC runs authorization checks on the basis of the lowest specified security level. FTAC users may reduce but not increase the values specified for them by the FTAC administrator.

MAX-LEVELS = *UNCHANGED
The security levels set in this admission set are to remain unchanged.

MAX-LEVELS = *STD
For this admission set, the values of the default admission set are valid. The admission set is deleted from the admission file. This is possible if the user ID has already been deleted.

MAX-LEVELS = <integer 0...100>
With this value, you can set a maximum security level for all six basic functions. The value 0 means that no file transfer is possible on this user ID until further notice (until the admission set is modified again).

MAX-LEVELS = *PARAMETERS(...)
With this structure, you can set a maximum security level for each of the basic functions.

OUTBOUND-SEND =
Sets the maximum security level for the basic function “outbound send”. The owner of the admission set can send files to all partner systems whose security level has this value or lower.

OUTBOUND-SEND = *UNCHANGED
The value for OUTBOUND-SEND remains unchanged.

OUTBOUND-SEND = *STD
For OUTBOUND-SEND, the value from the default admission set is used.

OUTBOUND-SEND = <integer 0..100>
For OUTBOUND-SEND, this maximum security level is entered in the admission set.

OUTBOUND-RECEIVE =
Sets the maximum security level for the basic function “outbound receive”. The owner of the admission set can receive files from all partner systems whose security level has this value or lower.

OUTBOUND-RECEIVE = *UNCHANGED
The value for OUTBOUND-RECEIVE remains unchanged.

OUTBOUND-RECEIVE = *STD
For OUTBOUND-RECEIVE, the value from the default admission set is used.

OUTBOUND-RECEIVE = <integer 0..100>
For OUTBOUND-RECEIVE, this maximum security level is entered in the admission set.

INBOUND-SEND =
Sets the maximum security level for the basic function “inbound send”. All partner systems with this security level or lower can request files from the owner of the admission set.

INBOUND-SEND = *UNCHANGED
The value for INBOUND-SEND remains unchanged.

INBOUND-SEND = *STD
For INBOUND-SEND, the value from the default admission set is used.

INBOUND-SEND = <integer 0..100>
For INBOUND-SEND, this maximum security level is entered in the admission set.

INBOUND-RECEIVE =
Sets the maximum security level for the basic function “inbound receive”. All partner systems with this security level or lower may send files to the owner of the admission set.

INBOUND-RECEIVE = *UNCHANGED
The value for INBOUND-RECEIVE remains unchanged.

INBOUND-RECEIVE = *STD
For INBOUND-RECEIVE, the value from the default admission set is used.

INBOUND-RECEIVE = <integer 0..100>
For INBOUND-RECEIVE, this maximum security level is entered in the admission set.

INBOUND-PROCESSING =
Sets the maximum security level for the basic function “inbound processing”. All partner systems which have this security level or lower may include follow-up processing in their system as part of an FT request.

INBOUND-PROCESSING = *UNCHANGED
The value for INBOUND-PROCESSING remains unchanged.

INBOUND-PROCESSING = *STD
For INBOUND-PROCESSING, the value from the default admission set is used.

INBOUND-PROCESSING = <integer 0..100>
For INBOUND-PROCESSING, this maximum security level is entered in the admission set.

INBOUND-MANAGEMENT =
Sets the maximum security level for the basic function “inbound file management”. All partner systems with this security level or lower may include the modification of file attributes and the querying of directories as part of their FT request.

INBOUND-MANAGEMENT = *UNCHANGED
The value for INBOUND-MANAGEMENT remains unchanged.

INBOUND-MANAGEMENT = *STD
For INBOUND-MANAGEMENT, the value from the default admission set is used.

INBOUND-MANAGEMENT = <integer 0..100>
For INBOUND-MANAGEMENT, this maximum security level is entered in the admission set.

Example

Dagobert Duck, the FTAC administrator of the Duck Bank, wishes set up the admission set for his employee Donald, such that Donald

– can send files to partner systems with the security level of 10 or lower (basic function “outbound send”),

– can request files from partner systems with the security level of 10 or lower (basic function “outbound receive”).

He wants all partner systems to be able send files to and request files from the user ID DONALD, so he sets the security level for INBOUND-SEND and INBOUND-RECEIVE to 100.

Dagobert does not wish to permit follow-up processing to be initiated from external partners, since he is too stingy to want to make his resources available to others. Therefore, he sets INBOUND-PROCESSING and INBOUND-FILEMANAGEMENT at 0. Since these values are set in the default admission set for the Duck Bank, these specifications are used for *STD. No FTAC password is defined.

The long form of the required command is as follows

```
/MODIFY-FT-ADMISSION-SET USER-IDENTIFICATION=DONALD,                -
/                        MAX-LEVELS=(OUTBOUND-SEND=10,             -
/                                   OUTBOUND-RECEIVE=10,           -
/                                   INBOUND-SEND=100,              -
/                                   INBOUND-RECEIVE=100,           -
/                                   INBOUND-PROCESSING=*STD,       -
/                                   INBOUND-MANAGEMENT=*STD)
```

A possible short form of this command would be

```
/MOD-FT-ADM DONALD,MAX-LEV=(10,10,100,100,*STD,*STD)
```

Command return codes

(SC2)	SC1	Maincode	Meaning
0	0	FTC0050	The set security level exceeds the administrator‘s limit and will remain invalid until the administrator’s limit is raised accordingly.
0	64	FTC0150	The authorization password is missing.
0	64	FTC0151	Only the administrator or owner is permitted to make this modification.
0	64	FTC0152	The user ID entered is not the user‘s own user ID.
0	64	FTC0175	The operand “NEW-PASSWORD” may not be entered for *STD.
0	64	FTC0176	The user ID entered does not exist in the system.
0	64	FTC0255	A system error occurred.

SC1/2 = Subcode 1/2 in decimal notation

For additional information, see section “Command return codes” on page 102

4.20 MODIFY-FT-INSTANCE Modify an *open*FT instance

Using this command, you can modify the characteristics of an instance (name, automatic start of *open*FT).

MODIFY-FT-INSTANCE may only be set up if *open*FT is not started in this instance, (NOT-STARTED is displayed in the SHOW-FT-INSTANCE command).

WARNING!

The instance may not be renamed; this is because system resources that contain the instance name may still be occupied even though *open*FT has been terminated. That is the case, for example, if requests with pre- or post-processing are still entered under this instance.

MODIFY-**FT**-**INST**ANCE
NAME = <alphanum-name 1...8> ,**NEW-NAME** = ***UNCHANGED** / <alphanum-name 1...8> ,**AUTO**MATIC-**START** = ***UNCHANGED** / ***ON** / ***OFF**

Operands

NAME = <alphanum-name 1..8>
Name of the *open*FT instance that is to be modified.

NEW-NAME = *UNCHANGED
The instance name remains unchanged.

NEW-NAME = <alphanum-name 1..8>
The new instance name. This name must be identical on all the computers on which this instance is to be used.

AUTOMATIC-START =
This is specified if, after loading the instance, *open*FT is automatically started in this instance.

AUTOMATIC-START = *UNCHANGED
The previous setting remains unchanged.

AUTOMATIC-START = *OFF
After loading the instance, *open*FT is not started.

AUTOMATIC-START = *ON
After each loading of the instance, a START-FT command is also implicitly executed in this instance. In this way, it is possible to work with *open*FT immediately after loading. All the components that are available for the standard instance are also started such as, for example, *open*FT-AC and *open*FT-FTAM.

Command return codes

(SC2)	SC1	Maincode	Meaning
83	32	CMD0221	Internal error
25	64	FTR1025	Instance does not exist.
26	64	FTR1026	Instance must not be modified.

SC1/2 = Subcode 1/2 in decimal notation
For additional information, see section “Command return codes” on page 102

4.21 MODIFY-FT-OPTIONS (FTMODOPT) Modify operating parameters

The MODIFY-FT-OPTIONS (or FTMODOPT) command is used to modify one or more operating parameters of the local system. The relationships between the different operating parameters are explained in section “Optimizing the operating parameters” on page 31.

After setting up an instance that is not working via the standard host, a host must be configured for this instance using the MODIFY-FT-OPTIONS command. Only then can *open*FT be started for the first time in this instance. Using MODIFY-FT-OPTIONS, the instance ID is also set.

After installation, the authentication check is deactivated. Using MODIFY-FT-OPTIONS it is possible to activate and (if necessary) deactivate the authentication check (PARTNER-CHECK parameter). This parameter can be individually configured for each partner system using MODIFY-FT-PARTNER. The expanded sender checking is only valid for partner systems that are not authenticated in the local system.

If, after installation, the security level at partners was not set with ADD-FT-PARTNER or MODIFY-FT-PARTNER then the following automatic mechanism applies (SECURITY-LEVEL=*BY-PARTNER-ATTRIBUTES):

- Authenticated partners are assigned the security level 10,
- Partners known in BCAM are assigned the security level 90,
- Partners accessed via their IP address are assigned the security level 100.

Any unspecified operating parameters remain unchanged. The current operating parameters can be queried at any time using the SHOW-FT-OPTIONS command (see page 263).

The MODIFY-FT-OPTIONS command is also used to switch the FT trace, SNMP traps and console traps on and off and to control FT logging.

MODIFY-**FT**-**OPT**IONS / **FTMODOPT**

PROCESS-**LIM**IT = ***UNCHA**NGED / <integer 1..32> / ***NONE**

,**CONN**ECTION-**LIM**IT = ***UNCHA**NGED / <integer 1..255>

,**REQ**UEST-**WAIT**-LEVEL = ***UNCHA**NGED

,**PACING** = ***UNCHA**NGED

,**TRANS**PORT-**UNIT**-SIZE = ***UNCHA**NGED / <integer 512..65535>

,**SEC**URITY-**LEV**EL = ***UNCHA**NGED / ***BY**-**PART**NER-**ATTR**IBUTES / <integer 1..100>

,**PART**NER-**CHECK** = ***UNCHA**NGED / ***STD** / ***TRANS**PORT-**ADDR**ESS

,**TRACE** = ***UNCHA**NGED / ***ON** / ***OFF** / ***PAR**AMETERS(...)

***PAR**AMETERS(...)

SWITCH = ***UNCHA**NGED / ***ON** / ***OFF**

,**PART**NER-**SELECTION** = ***UNCHA**NGED / ***ALL** / ***NONE** / list-poss(3): ***OPENFT** / ***FTAM** / ***FTP**

,**REQ**UEST-**SELECTION** = ***UNCHA**NGED / ***ALL** / list-poss(2): *ONLY-**SYNC** / *ONLY-**ASYNC** / *ONLY-**LOC**AL / *ONLY-**REM**OTE

,**OPT**IONS = ***UNCHA**NGED / ***NONE** / list-poss(1): ***NO-BULK**-DATA

,**LOGG**ING = ***UNCHA**NGED / ***SEL**ECT(...)

***SEL**ECT(...)

TRANSFER-**FILE** = ***UNCHA**NGED / ***OFF** / ***ON** / ***FAILURE**

,**FTAC** = ***UNCHA**NGED / ***ON** / ***REJECTED** / ***MODIFICATIONS**

,**MAX-INB**OUND-**REQ**UEST = ***UNCHA**NGED

,**REQ**UEST-**LIM**IT = ***UNCHA**NGED / <integer 2..32000>

,**MAX-REQUEST-LIFETIME** = ***UNCHA**NGED / ***UNLIM**ITED / <integer 1..400>

,**SNMP-TRAPS** = ***UNCHA**NGED / ***ALL** / ***NONE** / ***PAR**AMETERS(...)

***PAR**AMETERS(...)

SUBSYSTEM-**STATE** = ***UNCHA**NGED / ***OFF** / ***ON**

,**FT-STATE** = ***UNCHA**NGED / ***OFF** / ***ON**

,**PART**NER-**STATE** = ***UNCHA**NGED / ***OFF** / ***ON**

,**PARTNER-UNREACHABLE** = ***UNCHA**NGED / ***OFF** / ***ON**

,**REQUEST-QUEUE-STATE** = ***UNCHA**NGED / ***OFF** / ***ON**

,**TRANSFER-SUCCESS** = ***UNCHA**NGED / ***OFF** / ***ON**

,**TRANSFER-FAILURE** = ***UNCHA**NGED / ***OFF** / ***ON**

```
,CONSOLE-TRAPS = *UNCHANGED / *ALL / *NONE / *PARAMETERS(...)
    *PARAMETERS(...)
        SUBSYSTEM-STATE = *UNCHANGED / *OFF / *ON
        ,FT-STATE = *UNCHANGED / *OFF / *ON
        ,PARTNER-STATE = *UNCHANGED / *OFF / *ON
        ,PARTNER-UNREACHABLE = *UNCHANGED / *OFF / *ON
        ,REQUEST-QUEUE-STATE = *UNCHANGED / *OFF / *ON
        ,TRANSFER-SUCCESS = *UNCHANGED / *OFF / *ON
        ,TRANSFER-FAILURE = *UNCHANGED / *OFF / *ON
,HOST-NAME = *UNCHANGED / <alphanum-name 1..8>
,IDENTIFICATION = *UNCHANGED / <c-string 1..64 with-low> / <composed-name 1..64>
,KEY-LENGTH = *UNCHANGED / 0 / 768 / 1024
,OPENFT-APPLICATION = *UNCHANGED / *STD / <text 1..24>
,OPENFT-STD = *UNCHANGED / *STD /  <integer 1..65535>
,FTAM-APPLICATION = *UNCHANGED / *STD /  <text 1..40>
,FTP-PORT = *UNCHANGED / *NONE / *STD / <integer 1..65535>
,DYNAMIC-PARTNERS = *UNCHANGED / *OFF / *ON
```

Operands

PROCESS-LIMIT =
PROCESS-LIMIT is the maximum number of tasks that can be reserved simultaneously for the execution of file transfer requests.

PROCESS-LIMIT = *UNCHANGED
PROCESS-LIMIT is not changed, default value.

PROCESS-LIMIT = <integer 1..32>
PROCESS-LIMIT can have any value between 1 and 32.
The default value is 2.

PROCESS-LIMIT = *NONE
A server task is created for each new connection. PROCESS-LIMIT is therefore restricted by CONNECTION-LIMIT.

CONNECTION-LIMIT =
CONNECTION-LIMIT is the maximum number of transport connections that can be reserved for the execution of FT requests. This limit does not include file management requests and synchronous requests.
The maximum number of transport connections cannot be exceeded, not even if there are many high-priority file transfer requests to be executed. Since only one request can be processed at a time per transport connection, CONNECTION-LIMIT is also the maximum number of requests which a system can process simultaneously. One third of the transport connections defined by CONNECTION-LIMIT are reserved for requests from the remote system, and another third for requests submitted in the local system. The remaining third are available for both local and remote requests. This prevents locally submitted requests from blocking the system against requests from remote systems. If CONNECTION-LIMIT is less than 3, no transport connections are reserved.

CONNECTION-LIMIT = *UNCHANGED
The CONNECTION-LIMIT value is not changed, default value.

CONNECTION-LIMIT = <integer 1..255>
CONNECTION-LIMIT can have any value between 1 and 255.
Default value: 8

REQUEST-WAIT-LEVEL =
REQUEST-WAIT-LEVEL is now only supported for reasons of compatibility.

REQUEST-WAIT-LEVEL = *UNCHANGED
The value for REQUEST-WAIT-LEVEL is unchanged.

PACING =
PACING is now only supported for reasons of compatibility.

PACING = *UNCHANGED
The value of PACING is unchanged.

TRANSPORT-UNIT-SIZE =
specifies the maximum size of a transport unit in bytes.

TRANSPORT-UNIT-SIZE = *UNCHANGED
The current value size of a transport unit in bytes is unchanged.

TRANSPORT-UNIT-SIZE = <integer 512..65535>
TRANSPORT-UNIT-SIZE can assume any value between 512 and 65535.
The default value: 65535 bytes after installation.

It is recommended that you use this default value. TRANSPORT-UNIT-SIZE does not affect links with FTAM partners.

SECURITY-LEVEL =
This parameter need only be specified when FTAC functionality is used. An important part of the access protection functions provided by this product is based on the allocation of a security level to each partner. These security levels are designated using integers. The FT administrator can define a global value. This security level applies to all partner systems that are not explicitly assigned their own security levels when entered in the network description.

SECURITY-LEVEL = *UNCHANGED
The security level is unchanged.

SECURITY-LEVEL = *BY-PARTNER-ATTRIBUTES
If you set the operand to *BY-PARTNER-ATTRIBUTES then the security level is defined automatically. This setting assigns partners that are authenticated by *open*FT the security level 10. Partners that are known in BCAM (i.e. they are addressed via their BCAM names) are assigned the security level 90. All other partners are assigned security level 100. This setting is the default as of *open*FT V10.

SECURITY-LEVEL = <integer 1..100>
SECURITY-LEVEL can assume any value between 1 and 100. The default value is 1. If FTAC functionality is to be used, remember that 1 is the lowest level of security, offering the least protection. This is sufficient if you do not wish to further differentiate your remote systems; otherwise, a higher value should be defined. The allocation of different security levels is particularly meaningful if the authentication check is activated.

PARTNER-CHECK =
Is used to activate the extended authentication check. When using expanded sender checking, not only the partner identification is checked, but also the transport address. PARTNER-CHECK only affects *open*FT partners that are not authenticated in the local system (see the section “Authentication” on page 46). For FTAM partners, sender checking is enabled.

The globally set expanded sender checking can be modified for specific partners. See the operand PARTNER-CHECK for the ADD-FT-PARTNER and MODIFY-FT-PARTNER commands.

PARTNER-CHECK = *UNCHANGED
The existing value is retained.

PARTNER-CHECK = *STD
If dynamic partners are prohibited (DYNAMIC-PARTNERS=*OFF), a check is performed to determine whether the partner is entered in the partner list as a partner system, and only then will the file transfer be allowed. The default value on creating the request and network description file is *STD. You can output the settings with the SHOW-FT-OPTIONS command and the specification PARTNER-CHECK=*STD.
If dynamic partners are permitted (DYNAMIC-PARTNERS=*ON), transfers are also permitted from partners that are accessed via their address and/or are not entered in the partner list.

PARTNER-CHECK = *TRANSPORT-ADDRESS
Extended authentication check**.** In addition to checking whether the partner is entered in its own partner list as a partner system, it is checked whether the transport address under which the partner logs on matches the transport address entered in the partner list for the partner system.
If dynamic partners are permitted then this setting has no significance.

TRACE =
Switches FT trace functions on.

TRACE = *UNCHANGED
The existing FT trace functions remain unchanged.

TRACE = *ON
Switches the FT trace functions on.

TRACE = *OFF
Switches the FT trace functions off.

TRACE = *PARAMETERS(...)
Selects the option that is to be applied when writing the trace.

SWITCH =
Deactivates the FT trace functions for the selected partners.

SWITCH = *UNCHANGED
The previous value is unchanged.

SWITCH = *ON
Activates the FT trace functions.

SWITCH = *OFF
Deactivates the FT trace functions.

PARTNER-SELECTION =
Selects the partners that are to be monitored. The selection made here can be modified with the TRACE operand of the MODIFY-FT-PARTNER command.

PARTNER-SELECTION = *UNCHANGED
The previous value is unchanged.

PARTNER-SELECTION = *ALL
All the partners are selected for monitoring.

PARTNER-SELECTION = *NONE
No partner is selected for monitoring. Only those partners are monitored which have been selected for tracing with the TRACE operand of the MODIFY-FT-PARTNER command.

PARTNER-SELECTION = *OPENFT
All partners which are addressed via the *open*FT protocol are selected for monitoring.

PARTNER-SELECTION = *FTAM
All partners which are addressed via the FTAM protocol are selected for monitoring.

PARTNER-SELECTION = *FTP
All partners which are addressed via the FTP protocol are selected for monitoring.

REQUEST-SELECTION =
Selects the request types that are to be monitored.

REQUEST-SELECTION = *UNCHANGED
The previous value is unchanged.

REQUEST-SELECTION = *ALL
All the requests are selected for monitoring.

REQUEST-SELECTION = *ONLY-SYNC
All synchronous requests are selected for monitoring.

REQUEST-SELECTION = *ONLY-ASYNC
All asynchronous requests are selected for monitoring.

REQUEST-SELECTION = *ONLY-LOCAL
All locally submitted requests are selected for monitoring.

REQUEST-SELECTION = *ONLY-REMOTE
All remotely submitted requests are selected for monitoring.

OPTIONS =
Controls the options for the monitoring functions.

OPTIONS = *UNCHANGED
The previous value is unchanged.

OPTIONS = *NONE
No options are selected for the monitoring functions.

OPTIONS = *NO-BULK-DATA
If file contents (bulk data) are transferred with a protocol element and multiple trace records with the same protocol element occur in succession then only the first of these records is written to the trace file. This reduces the volume of the trace file.

LOGGING =
Switches FT and FTAC logging functions.

LOGGING = *UNCHANGED
The existing FT/FTAC logging functions remain unchanged.

LOGGING = *SELECT(...)
Switches on/off the FT logging function and controls the FTAC logging function.

TRANSFER-FILE = *UNCHANGED
The previous FT logging functions remain unchanged.

TRANSFER-FILE = *OFF
Switches the FT logging functions off.

TRANSFER-FILE = *ON
Switches the FT logging functions on.

TRANSFER-FILE = *FAILURE
Only errored requests are written to the logging file.

FTAC = *UNCHANGED
The previous FTAC logging functions remain unchanged.

FTAC = *ON
Switches the FTAC logging functions on.

FTAC = *REJECTED
All requests rejected by FTAC are logged.

FTAC = *MODIFICATIONS
All modifying requests are logged.

MAX-INBOUND-REQUEST =
MAX-INBOUND-REQUEST is now only supported for reasons of compatibility.

MAX-INBOUND-REQUEST = *UNCHANGED
The maximum number of inbound requests permitted remains unchanged.

REQUEST-LIMIT =
Changes the number of requests which can be saved in the request queue. Although it is logically possible to reduce the size of the request queue, this does not result in any memory being freed but only reduces the size of the internal queue. To free memory, it is necessary to end the FT subsystem, delete the request queue (SYSRQF) and then restart *open*FT.

REQUEST-LIMIT = *UNCHANGED
The previous value remains unchanged.

REQUEST-LIMIT = <integer 2..32000>
The maximum number of requests which can be saved in the request queue is changed to the value specified.

MAX-REQUEST-LIFETIME =
Limits the lifetime of FT requests in the request file. The maximum lifetime applies to inbound and outbound requests and is specified in days. The default value when a new request file is generated is *UNLIMITED.

The maximum lifetime does not apply to requests that have been transferred from an earlier request file as part of a version change. Such requests still have to be terminated using the CANCEL-FILE-TRANSFER command.

MAX-REQUEST-LIFETIME = *UNCHANGED
The previous value remains unchanged.

MAX-REQUEST-LIFETIME = *UNLIMITED
The lifetime of FT requests is unlimited.

MAX-REQUEST-LIFETIME = <integer 1..400>
The maximum lifetime for FT requests may have a value of between 1 and 400 days.

SNMP-TRAPS =
Activates or deactivates specific SNMP traps. SNMP traps are generated to indicate specific events which are routed by the FT subagent to an SNMP Management Station if one is in use. The default value on the generation of a new request file is *NONE.

SNMP-TRAPS = *UNCHANGED
The previous value is unchanged.

SNMP-TRAPS = *NONE
Deactivates all SNMP traps.

SNMP-TRAPS = *ALL
Activates all SNMP traps.

SNMP-TRAPS = *PARAMETERS(...)
Activates or deactivates selected SNMP traps. For further information, please refer to chapter “Operation” on page 29.

SUBSYSTEM-STATE =
Determines trap transmission on START-SUBSYSTEM / STOP-SUBSYSTEM and consequently indicates subsystem status.

SUBSYSTEM-STATE = *UNCHANGED
The previous value is unchanged.

SUBSYSTEM-STATE = *OFF
Deactivates the SUBSYSTEM-STATE trap.

SUBSYSTEM-STATE = *ON
Activates the SUBSYSTEM-STATE trap.

FT-STATE =
Determines trap transmission on START-FT / STOP-FT or abnormal FT termination.

FT-STATE = *UNCHANGED
The previous value is unchanged.

FT-STATE = *OFF
Deactivates the FT-STATE traps.

FT-STATE = *ON
Activates the FT-STATE traps.

PARTNER-STATE =
Determines trap transmission when the status of FT partners changes.

PARTNER-STATE = *UNCHANGED
The previous value is unchanged..

PARTNER-STATE = *OFF
Deactivates the PARTNER-STATE traps.

PARTNER-STATE = *ON
Activates the PARTNER-STATE traps.

PARTNER-UNREACHABLE =
Determines transmission of the trap that indicates if a partner cannot be accessed.

PARTNER-UNREACHABLE = *UNCHANGED
The previous value is unchanged.

PARTNER-UNREACHABLE = *OFF
Deactivates the “partner unreachable” trap.

PARTNER-UNREACHABLE = *ON
Activates the “partner unreachable” trap.

REQUEST-QUEUE-STATE =
Determines the transmission of traps when the request queue is more than 85% or less than 80% full.

REQUEST-QUEUE-STATE = *UNCHANGED
The previous value is unchanged.

REQUEST-QUEUE-STATE = *OFF
Deactivates the request queue fill level traps.

REQUEST-QUEUE-STATE = *ON
Activates the request queue fill level traps.

TRANSFER-SUCCESS =
Determines the transmission of the trap that indicates that an FT request has been successfully concluded.

TRANSFER-SUCCESS = *UNCHANGED
The previous value is unchanged.

TRANSFER-SUCCESS = *OFF
Deactivates the TRANSFER-SUCCESS trap.

TRANSFER-SUCCESS = *ON
Activates the TRANSFER-SUCCESS trap.

TRANSFER-FAILURE =
Determines the transmission of the trap that indicates that an FT request has been aborted.

TRANSFER-FAILURE = *UNCHANGED
The previous value is unchanged.

TRANSFER-FAILURE = *OFF
Deactivates the TRANSFER-FAILURE trap.

TRANSFER-FAILURE = *ON
Activates the TRANSFER-FAILURE trap.

CONSOLE-TRAPS =
Activates or deactivates console traps. By default, these trap messages are not displayed at the console. However, they are logged in the CONSLOG file. They can therefore cause storage problems on systems with high request volumes.
By default, the output of console traps is activated.

CONSOLE-TRAPS = *UNCHANGED
The previous value is unchanged.

CONSOLE-TRAPS = *ALL
The FTR03XX console messages are output by *open*FT. They always appear in the CONSLOG file. However, they are only output to the console if they are explicitly requested using the following command, e.g.:

```
/MOD-MSG-SUBSCRIPTION ADD-MSG-ID=(FTR0301,FTR0307,FTR0340,FTR0341)
```

CONSOLE-TRAPS = *NONE
The FTR03XX console messages are not output.

CONSOLE-TRAPS = *PARAMETERS(...)
Explicit specification of the events for which FTR03XX console messages are output.

SUBSYSTEM-STATE =
Controls the output of FTR03XX console messages concerning the status of the *open*FT subsystems.

SUBSYSTEM-STATE = *UNCHANGED
The previous value is unchanged.

SUBSYSTEM-STATE = *OFF
No FTR03XX console messages concerning the status of the *open*FT subsystem are output.

SUBSYSTEM-STATE = *ON
FTR03XX console messages concerning the status of the *open*FT subsystem are output.

FT-STATE =
Controls the output of FTR03XX console messages concerning the status of the *open*FT control process.

FT-STATE = *UNCHANGED
The previous value is unchanged.

FT-STATE = *OFF
No FTR03XX console messages concerning the status of the *open*FT control process are output.

FT-STATE = *ON
FTR03XX console messages concerning the status of the *open*FT control process are output.

PARTNER-STATE =
Controls the output of FTR03XX console messages concerning the status of the partner systems.

PARTNER-STATE = *UNCHANGED
The previous value is unchanged.

PARTNER-STATE = *OFF
No FTR03XX console messages concerning the status of partner systems are output.

PARTNER-STATE = *ON
FTR03XX console messages concerning the status of partner systems are output.

PARTNER-UNREACHABLE =
Controls the output of FTR03XX console messages if partner systems cannot be accessed.

PARTNER-UNREACHABLE = *UNCHANGED
The previous value is unchanged.

PARTNER-UNREACHABLE = *OFF
No FTR03XX console messages are output if partner systems cannot be accessed.

PARTNER-UNREACHABLE = *ON
FTR03XX console messages are output if partner systems cannot be accessed.

REQUEST-QUEUE-STATE =
Controls the output of FTR03XX console messages concerning the status of the request queue.

REQUEST-QUEUE-STATE = *UNCHANGED
The previous value is unchanged.

REQUEST-QUEUE-STATE = *OFF
No FTR03XX console messages concerning the status of the request queue are output.

REQUEST-QUEUE-STATE = *ON
FTR03XX console messages concerning the status of the request queue are output.

TRANSFER-SUCCESS =
Controls the output of FTR03XX console messages when a request is terminated successfully.

TRANSFER-SUCCESS = *UNCHANGED
The previous value is unchanged.

TRANSFER-SUCCESS = *OFF
No FTR03XX console messages are output if a request is terminated successfully.

TRANSFER-SUCCESS = *ON
FTR03XX console messages are output if a request is terminated successfully.

TRANSFER-FAILURE =
Controls the output of FTR03XX console messages when a request fails.

TRANSFER-FAILURE = *UNCHANGED
The previous value is unchanged.

TRANSFER-FAILURE = *OFF
No FTR03XX console messages are output if a request fails.

TRANSFER-FAILURE = *ON
FTR03XX console messages are output if a request fails.

HOST-NAME =
For using the *open*FT instance concept: Here you can set the BCAM host to which the transport system calls are made.

HOST-NAME = *UNCHANGED
The setting for the BCAM host remains unchanged.

HOST-NAME = <alphanum-name 1..8>
The name of the BCAM host via which the requests are processed. The result of this is that requests of an *open*FT instance are always processed via the same network address, irrespective of the real host. If an instance is to run on a virtual host, then the host name must be entered here before the first START-FT. Later, the host name should not be changed. It may not be changed if requests are present in the request file of this instance.

IDENTIFICATION =
Specifies the local instance ID of your *open*FT instance. With the aid of this instance ID, *open*FT partners as of V8.1 manage the resources for your *open*FT instance.

The instance ID must be unique, network-wide and must not be case-sensitive. An instance ID may consist of alphanumeric characters or special characters and may have a maximum length of 64 characters. It is advisable only to use the special characters “.”, “-”, “:” or “%”. The initial character must be alphanumeric or the special character “%”. The character “%” may only be used as an initial character. The character “.” must be followed by an alphanumeric character. For further details on assigning instance identifications, see section “Instance identification” on page 47.

When an instance is installed for the first time, the BCAM name of the real host under which their instance operates is entered as the default value. If another identification is to be used for operation then this must be configured with MODIFY-FT-OPTIONS.

IDENTIFICATION = *UNCHANGED
The instance ID remains unchanged.

IDENTIFICATION = <c-string with-low 1..64> / <composed-name 1..64>
The instance ID is set to this value.

KEY-LENGTH =
Defines the length of the RSA key used for encryption. This key is used only to encrypt the AES key which is agreed on between the partners (or the DES key up to and including *open*FT V7.0). *open*FT uses the AES key to encrypt the request description data and possibly also the file contents.

KEY-LENGTH = *UNCHANGED
The previous value is unchanged.

KEY-LENGTH = 0
Explicitly disables encryption.

KEY-LENGTH = 768 / 1024
Specifies the key length in bits.

OPENFT-APPLICATION =
This option allows you to specify a port number other than the default for the local *open*FT server.
Use this function carefully, as changing the port number from the default value will make it more difficult for the *open*FT partners to address the local system!

OPENFT-APPLICATION = *UNCHANGED
The previous value is unchanged.

OPENFT-APPLICATION = *STD
The port number is set to the default value 1100.

OPENFT-APPLICATION = <text 1..24>
Specifies a valid port number.

OPENFT-STD =
This option allows you to specify a port number other than the default when addressing *open*FT partners via their host names.
Use this function carefully, as changing the port number from the default value means that it will no longer be possible to reach *open*FT partners which use the default port number and are addressed via the host name!

OPENFT-STD = *UNCHANGED
The previous value is unchanged.

OPENFT-STD = *STD
The port number is set to the default value 1100.

OPENFT-STD = <integer 1..65535>
Specifies a valid port number.

FTAM-APPLICATION =
This option allows you to specify a port number other than the default for the local FTAM server.
Use this function carefully, as changing the port number from the default value will make it more difficult for the FTAM partners to address the local system!

FTAM-APPLICATION = *UNCHANGED
The previous value is unchanged.

FTAM-APPLICATION = *STD
The port number is set to the default value 4800.

FTAM-APPLICATION = <text 1..40>
Specifies a valid port number.

FTP-PORT =
This option allows you to specify the port number used by FTP.

FTP-PORT = *UNCHANGED
The previous value is unchanged.

FTP-PORT = *NONE
No port number is defined. The FTP server is deactivated, i.e. it cannot accept any inbound FTP requests.

FTP-PORT = *STD
The port number is set to the default value 21.

FTP-PORT = <integer 1..65535>
Specifies a valid port number.

DYNAMIC-PARTNERS =
Specifies whether dynamic partners are permitted.

DYNAMIC-PARTNERS = *UNCHANGED
The previous value is unchanged.

DYNAMIC-PARTNERS = *ON
Dynamic partners are permitted. This means that partner systems can also be accessed via their addresses and that transfer requests are also permitted with partners which are not entered in the partner list.

DYNAMIC-PARTNERS = *OFF
Dynamic partners are not permitted. This means that it is only possible to access partner systems which are entered in the partner list and are addressed via the partner name. Transfer requests with partners which are not entered in the partner list are not permitted.

Command return codes

(SC2)	SC1	Maincode	Meaning
83	32	CMD0221	Internal error.
87	32	CMD0221	No space left on device for internal files.
33	64	FTR1033	The public key files could not be updated.
35	64	FTR1035	User not authorized for this command.

SC1/2 = Subcode 1/2 in decimal notation
For additional information, see section “Command return codes” on page 102

Example

The maximum number of tasks to be executed in parallel is to be 3 and the maximum number of transport connections to be set up is to be 10:

```
/MODIFY-FT-OPTIONS PROCESS-LIMIT=3,CONNECTION-LIMIT=10
```

4.22 MODIFY-FT-PARTNER (FTMODPTN) Modify partner properties in the partner list

This command can be used to modify the characteristics of a partner that is already entered in the partner list. When changing the partner address, please note that an *open*FT partner cannot be changed to an FTAM partner and vice versa.

MODIFY-**FT**-**PART**NER / **FTMODPTN**

PARTNER = ***ALL** / <text 1..200 with-low>

,**STATE** = ***UNCHA**NGED / ***ACTIVE**(...) / ***DEACT**

*ACTIVE(...)
| **AUTOMATIC-DEACT** = ***NO** / ***YES**

,**SECURITY-LEVEL** = ***UNCHA**NGED / ***STD** / ***BY-PART**NER-**ATTR**IBUTES / <integer 1..100>

,**PARTNER-ADDRESS** = ***UNCHA**NGED / <text 1..200 with-low>

,**TRACE** = ***UNCHA**NGED / ***BY-FT-OPT**IONS / ***ON** / ***OFF**

,**IDENTIFICATION** = ***UNCHA**NGED/ ***STD** / <composed-name 1..64> / <c-string 1..64 with-low>

,**SESSION-ROUTING-INFO** = ***UNCHA**NGED / ***NONE** / ***ID**ENTIFICATION / <alphanum-name 1..8>

,**PARTNER-CHECK** = ***UNCHANGED** / ***BY-FT-OPT**IONS / ***STD** / ***TRANS**PORT-**ADDR**ESS

,**AUTH-MANDATORY** = ***UNCHANGED** / ***NO** / ***YES**

Operands

PARTNER =
Specifies the partner system or systems.

PARTNER = *ALL
The specified changes are to be implemented for all partner systems defined in the partner list. This specification is only meaningful in conjunction with the operands STATE, SECURITY-LEVEL and TRACE. If the value of SECURITY-LEVEL is modified and if, simultaneously, dynamic partners are permitted, the value of the security level for dynamic partners remains unchanged and as set in the *open*FT options.

PARTNER = <text 1..200 with-low>
Specifies either the name of the partner system from the partner list or the address of the partner system (see section “Specifying partner addresses” on page 39).

STATE =
You can use this operand to control FT requests issued locally which are directed to a particular remote system.

STATE = *UNCHANGED
The state is unchanged.

STATE = *ACTIVE(...)
Deactivated FT requests to a remote system, which were issued locally are released again for processing.

AUTOMATIC-DEACT =
Defines if repeated attempts to establish a connection with a partner system should result in a deactivation of the partner system after a certain number of attempts.

AUTOMATIC-DEACT = *NO
Unsuccessful attempts to establish a connection with this partner do not lead to its deactivation.

AUTOMATIC-DEACT = *YES
Repeated unsuccessful attempts to establish a connection with this partner lead to its deactivation. If you wish to run file transfers with this partner afterwards, it must be explicitly reactivated.

STATE = *DEACT
FT requests to a remote system, which were issued locally are accepted, but will only be executed when the remote system is reactivated.

SECURITY-LEVEL =
This operand assigns a security level to a remote system.

SECURITY-LEVEL = *UNCHANGED
The value is unchanged.

SECURITY-LEVEL = *STD
If you set this operand to *STD, a standard security level is assigned to the remote system. This standard security level is defined using the MODIFY-FT-OPTIONS command.

SECURITY-LEVEL = *BY-PARTNER-ATTRIBUTES
If you set the operand to *BY-PARTNER-ATTRIBUTES then the security level is defined automatically. This setting assigns partners that are authenticated by *open*FT the security level 10. Partners that are known in BCAM (i.e. they are addressed via their BCAM names) are assigned the security level 90. All other partners are assigned security level 100.

SECURITY-LEVEL = <integer 1..100>
must be specified if you want to use FTAC functionality and assign a security level to an individual remote system.

PARTNER-ADDRESS =
is the address of the remote system.

PARTNER-ADDRESS = *UNCHANGED
The address remains unchanged.

PARTNER-ADDRESS = <text 1..200 with-low>
Specifies a new address for the partner system. For details on the address format, see section “Specifying partner addresses” on page 39.

TRACE =
Defines the trace setting for *open*FT partner systems. Trace entries are not generated unless TRACE=*ON is set in MODIFY-FT-OPTIONS.

TRACE = *UNCHANGED
The current trace setting is unchanged.

TRACE = *BY-FT-OPTIONS
The trace settings specified in the MODIFY-FT-OPTIONS command are used.

TRACE = *ON
Switches the trace on if tracing is activated for this partner type in the global settings (MODIFY-FT-OPTIONS). In this case, the request-specific settings are taken over from MODIFY-FT-OPTIONS.

TRACE = *OFF
For connections to this partner system, only those trace entries which it is technically impossible to suppress are generated. Trace entries which it is technically impossible to suppress are those which are generated before *open*FT (BS2000) identifies the partner system.

IDENTIFICATION =
The network-wide, unique ID of the *open*FT instance in the partner system.

IDENTIFICATION = *UNCHANGED
The ID remains unchanged.

IDENTIFICATION = *STD
The partner address is used as the identification.

IDENTIFICATION = <composed-name 1..64> / <c-string 1..64 with-low>
The network-wide, unique instance ID of the *open*FT instance in the partner system. It is set by the FT administrator of the partner system (for example in BS2000, by using MODIFY-FT-OPTIONS IDENTIFICATION=, in UNIX systems or Windows, by using ftmodo -id). The uniqueness of this ID must be based on something other than case-sensitivity. An instance ID may be comprised of alphanumeric characters or special characters. It is advisable to use only the special characters “.”, “-”, “:” or “%”. The initial character must be alphanumeric or the special character “%”. The “%” character may only be used as an initial character. An alphanumeric character must follow the “.” character. For more details on assigning instance identifications, see page 47.

You should always specify the instance identification of the partner system explicitly. Thus the performance is improved.

SESSION-ROUTING-INFO =
If the partner system is addressed via IDENTIFICATION, but is only accessible via a go-between instance (e.g. an *open*FTIF gateway), specify here the address information, which the go-between instance will use for re-routing. This is also necessary, for example, for partner systems using *open*FT for OS/390 and z/OS, dependent on TRANSIT coupling.

SESSION-ROUTING-INFO = *UNCHANGED
The setting remains unchanged.

SESSION-ROUTING-INFO = *NONE
No routing information is used.

SESSION-ROUTING-INFO = <alphanum-name 1..8>
Address of the partner in the gateway. In the case of *open*FTIF, this entry must be present in lowercase in the gateway computer's TNS.

SESSION-ROUTING-INFO = *IDENTIFICATION
Connections to the partner are re-routed via a gateway that supports IDENTIFICATION as the address information.

PARTNER-CHECK =
With this, you can set (for the partner) an allocation for sender checking that deviates from the global settings. PARTNER-CHECK is only effective for *openFT* partners that are not authenticated in the local system (see the section "Authentication" on page 46). For FTAM partners, sender checking is essentially enabled.

PARTNER-CHECK = *UNCHANGED
The set value remains unchanged.

PARTNER-CHECK = *BY-FT-OPTIONS
The global settings are valid for the partner.

PARTNER-CHECK = *STD
Disable the expanded sender checking. The transport address of the partner is not checked, even if the expanded sender checking is globally enabled (see the MODIFY-FT-OPTIONS command).

PARTNER-CHECK = *TRANSPORT-ADDRESS
Enables expanded sender checking. The transport address is checked, even if the expanded sender checking is globally disabled (see the MODIFY-FT-OPTIONS command). If the transport address under which the partner is reporting does not correspond to the entry in the partner list, the request is rejected.

AUTH-MANDATORY =
This option allows you to force the authentication of a partner.

AUTH-MANDATORY = *UNCHANGED
The set value is unchanged.

AUTH-MANDATORY = *NO
Authentication is not forced, i.e. this partner is not restricted with regard to authentication.

AUTH-MANDATORY = *YES
Authentication is forced, i.e. this partner may only work with authentication.

Command return codes

(SC2)	SC1	Maincode	Meaning
198	1	CMD0202	Invalid parameter value.
83	32	CMD0221	Internal error.
35	64	FTR1035	User not authorized for this command.
43	64	FTR1043	Partner with same attribute already exists in partner list.
44	64	FTR1044	Maximum number of partners exceeded.
45	64	FTR1045	Partner name not found in partner list.
46	64	FTR1046	Modification of partner protocol type not possible

SC1/2 = Subcode 1/2 in decimal notation
For additional information, see section “Command return codes” on page 102

Example 1

The SECURITY-LEVEL for the partner system TEST is set to 99:

```
/MODIFY-FT-PARTNER PARTNER=TEST,SECURITY-LEVEL=99
```

Example 2:

The port number for partner WINDOWS (host name = winhost2) is set to 1100:

```
/MODIFY-FT-PARTNER PARTNER=WINDOWS,PARTNER-ADDRESS=winhost2:1100
```

4.23 MODIFY-FT-PROFILE Modify admission profile

Prerequisite for using this command is the use of *open*FT-AC .
The command MODIFY-FT-PROFILE can be used by any FTAC user to modify his admission profile. In a privileged admission profile, an FTAC user can only modify the operands TRANSFER-ADMISSION and PRIVILEGED.

The FTAC administrator can privilege foreign admission profiles with the operand PRIVILEGED.

MODIFY-**FT-PROF**ILE

NAME = ***ALL** / <alphanum-name 1..8>

,**PASSWORD** = ***NONE** / <c-string 1..8 with-low> / <x-string 1..16> / ***SEC**RET

,**SELECT-PARAMETER** = ***OWN** / ***PAR**AMETERS(...)

***PAR**AMETERS(...)
- **TRANSFER-ADMISSION** = ***ALL** / ***NOT-SPEC**IFIED / <alphanum-name 8..32> / <c-string 8..32 with-low> / <x-string 15..64> / ***SEC**RET
- ,**OWNER-IDENTIFICATION** = ***OWN** / ***ALL** / <name 1..8>

,**NEW-NAME** = ***OLD** / <alphanum-name 1..8>

,**TRANSFER-ADMISSION** = ***UNCH**ANGED / ***NOT-SPEC**IFIED / ***OLD-ADM**ISSION(...) / <alphanum-name 8..32>(...) / <c-string 8..32 with-low>(...) / <x-string 15..64>(...) / ***SEC**RET

***OLD-ADM**ISSION(...)
- **VALID** =*__UNCH__ANGED/ ***YES** / ***NO**
- ,**USAGE** = *__UNCH__ANGED / ***PRIVATE** / ***PUB**LIC
- ,**EXPIRATION-DATE** = *__UNCH__ANGED / ***NOT-REST**RICTED / <date 8..10>

<alphanum-name 8..32>(...) / <c-string 8..32 with-low>(...) / <x-string 15..64>(...)
- **VALID** = ***YES** / ***NO** / ***UNCH**ANGED
- ,**USAGE** = ***PRIVATE** / ***PUB**LIC / ***UNC**HANGED
- ,**EXPIRATION-DATE** = ***NOT-REST**RICTED / <date 8..10> / ***UNCH**ANGED

,**PRIVILEGED** = ***UNCH**ANGED / ***NO** / ***YES**

,**IGNORE-MAX-LEVELS** = ***UNCH**ANGED / ***NO** / ***YES** / ***PAR**AMETERS(...)

***PAR**AMETERS(...)
- **OUTBOUND-SEND** = ***UNCH**ANGED / ***NO** / ***YES**
- ,**OUTBOUND-RECEIVE** = ***UNCH**ANGED / ***NO** / ***YES**
- ,**INBOUND-SEND** = ***UNCH**ANGED / ***NO** / ***YES**
- ,**INBOUND-RECEIVE** = ***UNCH**ANGED / ***NO** / ***YES**
- ,**INBOUND-PROCESSING** = ***UNCH**ANGED / ***NO** / ***YES**
- ,**INBOUND-MANAGEMENT** = ***UNCH**ANGED / ***NO** / ***YES**

```
,USER-ADMISSION = *UNCHANGED / *OWN / *PARAMETERS(...)

  *PARAMETERS(...)
    | USER-IDENTIFICATION = *OWN / <name 1..8>
    | ,ACCOUNT = *OWN / *FIRST / *NOT-SPECIFIED / <alphanum-name 1..8>
    | ,PASSWORD = *OWN / *NOT-SPECIFIED / <c-string 1..8> / <c-string 9..32> / <x-string 1..16> /
                  *NONE / *SECRET

,INITIATOR = *UNCHANGED / list-poss(2): *REMOTE / *LOCAL

,TRANSFER-DIRECTION = *UNCHANGED / *NOT-RESTRICTED / *FROM-PARTNER / *TO-PARTNER

,PARTNER = *UNCHANGED / *NOT-RESTRICTED / *ADD(...) / *REMOVE(...) /
                  list-poss(50): <text 1..200 with-low>

  *ADD(...)
    | NAME = list-poss(50): <text 1..200 with-low>

  *REMOVE(...)
    | NAME = list-poss(50): <text 1..200 with-low>

,MAX-PARTNER-LEVEL = *UNCHANGED / *NOT-RESTRICTED / <integer 0..100>

,FILE-NAME = *UNCHANGED / *NOT-RESTRICTED / <filename 1..54> / <c-string 1..512 with-low> /
             *EXPANSION(...) / *LIBRARY-ELEMENT(...) / *POSIX(NAME=<posix-pathname 1..219>)

  *EXPANSION(...)
    | PREFIX = <filename 1..53> / <partial-filename 2..53> / <c-string 1..511 with-low>

  *LIBRARY-ELEMENT(...)
    | LIBRARY = *UNCHANGED / *NOT-RESTRICTED / <filename 1..54> / *EXPANSION(...)
    |   *EXPANSION(...)
    |     | PREFIX = <filename 1..53> / <partial-filename 2..53>
    | ,ELEMENT = *UNCHANGED / *NOT-RESTRICTED /
    |                <composed-name 1..64 with-under>(...) / *EXPANSION(...)
    |    <composed-name 1..64 with-under>(...)
    |     | VERSION = *STD / <text 1..24>
    |   *EXPANSION(...)
    |     | PREFIX = <composed-name 1..63 with-under> / <partial-filename 2..63>
    | ,TYPE = *UNCHANGED / *NOT-RESTRICTED / <name 1..8>

,FILE-PASSWORD = *UNCHANGED / *NOT-RESTRICTED / *NONE / <c-string 1...4> / <x-string 1...8> /
                 <integer -2147483648...2147483647> / *SECRET

,PROCESSING-ADMISSION = *UNCHANGED / *SAME / *NOT-RESTRICTED / *PARAMETERS(...)

  *PARAMETERS(...)
    | USER-IDENTIFICATION = *SAME / *NOT-RESTRICTED / <name 1..8>
    | ,ACCOUNT = *SAME / *NOT-RESTRICTED/<alphanum-name 1..8>
    | ,PASSWORD = *SAME / *NOT-RESTRICTED / *NONE / <c-string 1..8> / <c-string 9..32> /
                                            <x-string 1..16> / *SECRET
```

```
,SUCCESS-PROCESSING = *UNCHANGED / *NOT-RESTRICTED / *NONE / <c-string 1..1000 with-low> /
                      *EXPANSION(...)
    *EXPANSION(...)
        PREFIX = *UNCHANGED / *NOT-RESTRICTED / <c-string 1..999 with-low>
        ,SUFFIX = *UNCHANGED / *NOT-RESTRICTED / <c-string 1..999 with-low>
,FAILURE-PROCESSING = *UNCHANGED / *NOT-RESTRICTED / *NONE / <c-string 1..1000 with-low> /
                      *EXPANSION(...)
    *EXPANSION(...)
        PREFIX = *UNCHANGED / *NOT-RESTRICTED / <c-string 1..999 with-low>
        ,SUFFIX = *UNCHANGED / *NOT-RESTRICTED / <c-string 1..999 with-low>
,WRITE-MODE = *UNCHANGED / *NOT-RESTRICTED / *NEW-FILE / *REPLACE-FILE / *EXTEND-FILE
,FT-FUNCTION = *UNCHANGED / *NOT-RESTRICTED / list-poss(4): *TRANSFER-FILE /
               *MODIFY-FILE-ATTRIBUTES / *READ-DIRECTORY/ *FILE-PROCESSING
,USER-INFORMATION = *UNCHANGED / *NONE / <c-string 1..100 with-low>
,DATA-ENCRYPTION = *UNCHANGED / *NOT-RESTRICTED / *NO / *YES
```

Operands

NAME =
With NAME, you determine the name of the admission profile to be modified.

NAME = *ALL
Use this to modify all your admission profiles at the same time.

NAME = <alphanum-name 1..8>
Use this to modify the admission profile with this name.

PASSWORD =
With PASSWORD, you enter the FTAC password which authorizes you to use FTAC commands on your user ID, if such a password has been defined in your admission set.

PASSWORD = *NONE
No FTAC password is required.

PASSWORD = <c-string 1..8 with-low> / <x-string 1..16>
This FTAC password is required.

PASSWORD = *SECRET
The system prompts you to enter the password. However, it does not appear on the screen.

SELECT-PARAMETER =
With SELECT-PARAMETER, you can specify a transfer admission. You will then modify the admission profile which has this transfer admission.

SELECT-PARAMETER = *OWN
Use this to modify your own admission profile.

SELECT-PARAMETER = *PARAMETERS(...)
With this structure, you can specify the selection criteria for the profiles which you wish to modify.

TRANSFER-ADMISSION =
Entering the TRANSFER-ADMISSION here makes it a selection criterion for the admission profiles which you wish to modify.

TRANSFER-ADMISSION = *ALL
All your admission profiles are to be modified, irrespective of the transfer admission.

TRANSFER-ADMISSION = *NOT-SPECIFIED
Only admission profiles without a defined transfer admission are to be modified.

TRANSFER-ADMISSION = <alphanum-name 8..32> /
<c-string 8..32 with-low> / <x-string 15..64>
The admission profile with this transfer admission is to be modified.

TRANSFER-ADMISSION = *SECRET
The system prompts you to enter the transfer admission. However, it does not appear on the screen.

OWNER-IDENTIFICATION =
With OWNER-IDENTIFICATION, you can use the owner of an admission profile as a selection criterion for access to a profile to be modified.

OWNER-IDENTIFICATION = *OWN
You wish to modify your own admission profile.

OWNER-IDENTIFICATION = *ALL
The FTAC administrator can access the profiles of all users. The FTAC user is not permitted to make this entry.

OWNER-IDENTIFICATION = <name 1..8>
The FTAC user can enter only his own user ID here, the FTAC administrator can enter any user ID.

NEW-NAME =
With NEW-NAME, you give your admission profile a new name (or not).

NEW-NAME = *OLD
The name of the admission profile remains unchanged.

NEW-NAME = <alphanum-name 1..8>
This is to be the new name of the admission profile. This name must be unique among all the admission profiles on your user ID. If an admission profile with this name already exists, FTAC rejects the command with the following message:

```
FTC0100   FT profile already exists.
```

The command SHOW-FT-PROFILE (see page 279ff) can be used to obtain information on the already existing name. For this information, it suffices to enter SHOW-FT-PROFILE without parameters.

TRANSFER-ADMISSION =
With TRANSFER-ADMISSION, you can modify the transfer admission which is associated with an admission profile. You must ensure that the transfer admission is unique within your *open*FT system. If the transfer admission which you have selected already exists, FTAC rejects the command with the following message:

```
FTC0101   Transfer admission already exists.
```

The FTAC administrator can also allocate an access admission here if he/she modifies the admissions profile of any user ID. To do this, however, the FTAC administrator must specify the complete USER-ADMISSION for the affected user ID (USER-IDENTIFICTATION, ACCOUNT, and PASSWORD).

TRANSFER-ADMISSION = *UNCHANGED
The transfer admission remains unchanged.

TRANSFER-ADMISSION = *NOT-SPECIFIED
No transfer admission is set and any existing transfer admissions are made invalid. This blocks the profile.

TRANSFER-ADMISSION = *OLD-ADMISSION(...)
The transfer admission itself remains unchanged. The options, however, can be changed, as opposed to with the entry TRANSFER-ADMISSION=*UNCHANGED. The description of the values in brackets (VALID=, USAGE= and EXPIRATION-DATE=) follow.

VALID = *YES
The transfer admission is valid.

VALID = *NO
The transfer admission is not valid. The profile can be blocked with this entry.

VALID = *UNCHANGED
The value remains unchanged.

USAGE = *PRIVATE
Access to your profile is denied for security reasons whenever another user ID attempts to set for a second time the TRANSFER-ADMISSION which has already been used by you.

USAGE = *PUBLIC
Access to your profile is not denied if another user happens to “discover” your TRANSFER-ADMISSION. “Discovery” means that another user ID attempted to specify the same TRANSFER ADMISSION twice. This is rejected for security reasons.

USAGE = *UNCHANGED
The value remains unchanged.

EXPIRATION-DATE = *NOT-RESTRICTED
The use of this transfer admission is not restricted with respect to time.

EXPIRATION-DATE = <date 8..10>
The use of the transfer admission is only possible until the given date (max. 19.01. 2038). The entry must be made in the form *yyyy-mm-dd* or *yy-mm-dd*.

EXPIRATION-DATE = *UNCHANGED
The value remains unchanged.

TRANSFER-ADMISSION = <alphanum-name 8..32>(...) / <c-string 8..32 with-low>(...) / <x-string 15..64>(...)
The character string must be entered as transfer admission in the transfer request. The alphanumeric input is always stored in lowercase letters.

VALID = *YES
The transfer admission is valid.

VALID = *NO
The transfer admission is not valid. The profile can be blocked with this entry.

VALID = *UNCHANGED
The value remains unchanged.

USAGE = *PRIVATE
Access to your profile is denied for security reasons whenever another user ID attempts to set for a second time the TRANSFER-ADMISSION which has already been used by you.

USAGE = *PUBLIC
Access to your profile is not denied if another user happens to “discover” your TRANSFER-ADMISSION. “Discovery” means that another user ID attempted to specify the same TRANSFER ADMISSION twice. This is rejected for security reasons.

USAGE = *UNCHANGED
The value remains unchanged.

EXPIRATION-DATE = *NOT-RESTRICTED
The use of this transfer admission is not restricted with respect to time.

EXPIRATION-DATE = <date 8..10>
The use of the transfer admission is only possible until the given date (max. 19.01.2038). The entry must be made in the form *yyyy-mm-dd* or *yy-mm-dd*.

EXPIRATION-DATE = *UNCHANGED
The value remains unchanged.

TRANSFER-ADMISSION = *SECRET
The system prompts you to input the transfer admission; however, this does not appear on the screen. The operands VALID, USAGE and EXPIRATION-DATE can also be secretly entered in this case.

PRIVILEGED =
With PRIVILEGED, the FTAC can privilege the admission profile of any FTAC user.
FT requests which are processed with a privileged status are not subject to the restrictions for MAX-ADM-LEVEL in the admission set.
The FTAC user can only reverse any privileged status given.

PRIVILEGED = *UNCHANGED
The status of this admission profile remains unchanged.

PRIVILEGED = *NO
With *NO, you can reverse the privileged status.

PRIVILEGED = *YES
With *YES, the FTAC administrator can give an admission profile privileged status.

IGNORE-MAX-LEVELS =
With IGNORE-MAX-LEVELS, you can determine for which of the six basic functions the restrictions of the admission set should be ignored. The user's MAX-USER-LEVELS can be exceeded in this way. The MAX-ADM-LEVELS in the admission set can only be effectively exceeded with an admission profile which has been designated as privileged by the FTAC administrator. The FTAC user can set up an admission profile for himself for special tasks (e.g. sending a certain file to a partner system with which he normally is not allowed to conduct a file transfer), which allows him to exceed the admission set. This profile must be explicitly given privileged status by the FTAC administrator.
If you enter IGNORE-MAX-LEVELS=*YES, the settings for all the basic functions are ignored. If you wish to ignore the admission set for specific basic functions, you need to do this with the operands explained later in the text.

The following table shows which partial components of the file management can be used under which conditions:

Inbound file management function	Setting in admission set/extension in profile
Show file attributes	Inbound sending (IBS) permitted
Modify file attributes	Inbound receiving (IBR) **and** Inbound file management (IBF) permitted
Rename files	Inbound receiving (IBR) **and** Inbound file management (IBF) permitted
Delete files	Inbound receiving (IBR) permitted **and** write rule = overwrite in profile
Show directories	Inbound file management (IBF) permitted **and** direction = to partner in profile
Create, rename, delete directories	Inbound file management (IBF) permitted **and** direction = from partner in profile

IGNORE-MAX-LEVELS = *UNCHANGED
With this admission profile, you can access the same security levels as before the modification (unless you have reversed the privileged status with PRIVILEGED=*NO).

IGNORE-MAX-LEVELS = *NO
FT requests which are processed with the admission profile are subject to the restrictions of the admission set.

IGNORE-MAX-LEVELS = *YES
*YES allows you to communicate with partner systems whose security level exceeds the specifications of the admission set. If your profile does not have privileged status, you can only disregard the MAX-USER-LEVELS in the admission set, not the MAX-ADM-LEVELS. The current MAX-USER-LEVELS and MAX-ADM-LEVELS settings can be accessed using the command SHOW-FT-ADMISSION-SET (see example on page 245).

IGNORE-MAX-LEVELS = *PARAMETERS(...)

OUTBOUND-SEND = *UNCHANGED
The maximum security level which can be reached with the basic function “outbound send” remains unchanged.

OUTBOUND-SEND = *NO
The maximum security level which can be reached with the basic function “outbound send” is determined by the admission set.

OUTBOUND-SEND = *YES
For the basic function “outbound send”, you can use this admission profile to disregard the MAX-USER-LEVELS. If your profile is privileged, you are also not held to the restrictions of the MAX-ADM-LEVELS.

OUTBOUND-RECEIVE = *UNCHANGED
The maximum security level which can be reached with the basic function “outbound receive” remains unchanged.

OUTBOUND-RECEIVE = *NO
The maximum security level which can be reached with the basic function “outbound receive” is determined by the admission set.

OUTBOUND-RECEIVE = *YES
For the basic function “outbound receive”, you can use this admission profile to disregard the MAX-USER-LEVELS. If your profile is privileged, you are also not held to the restrictions of the MAX-ADM-LEVELS.

INBOUND-SEND = *UNCHANGED
The maximum security level which can be reached with the basic function “inbound send” remains unchanged.

INBOUND-SEND = *NO
The maximum security level which can be reached with the basic function “inbound send” is determined by the admission set.

INBOUND-SEND = *YES
For the basic function “inbound send”, you can use this admission profile to disregard the MAX-USER-LEVELS. If your profile is privileged, you are also not held to the restrictions of the MAX-ADM-LEVELS. In addition, the partial component “display file attributes” of the basic function “inbound file management” can be used.

INBOUND-RECEIVE = *UNCHANGED
The maximum security level which can be reached with the basic function “inbound receive” remains unchanged.

INBOUND-RECEIVE = *NO
The maximum security level which can be reached with the basic function “inbound receive” is determined by the admission set.

INBOUND-RECEIVE = *YES
With this profile, you can disregard your settings for “inbound receive” in the MAX-USER-LEVELS. If your profile is privileged, you are also not held to the restrictions of the MAX-ADM-LEVELS. In addition, the following partial components of the basic function “inbound file management” can be used:
– delete files, as long as the file attributes are set accordingly,
– modify file attributes, if the basic function “inbound file management” was admitted in the admission set or in the admission profile.

INBOUND-PROCESSING = *UNCHANGED
The maximum security level which can be reached with the basic function “inbound processing” remains unchanged.

INBOUND-PROCESSING = *NO
The maximum security level which can be reached with the basic function “inbound processing” is determined by the admission set.

INBOUND-PROCESSING = *YES
For the basic function “inbound processing”, you can use this admission profile to disregard the MAX-USER-LEVELS. If your profile is privileged, you are also not held to the restrictions of the MAX-ADM-LEVELS.

INBOUND-MANAGEMENT = *UNCHANGED
The maximum security level which can be reached with the basic function “inbound file management” remains unchanged.

INBOUND-MANAGEMENT = *NO
The maximum security level which can be reached with the basic function “inbound file management” is determined by the admission set.

INBOUND-MANAGEMENT = *YES
For the basic function “inbound file management”, you can use this admission profile to disregard the MAX-USER-LEVELS. If your profile is privileged, you are also not held to the restrictions of the MAX-ADM-LEVELS. The partial component “modify file attributes” of the basic function “inbound file management” only functions if the basic function “inbound receive” was admitted in the admission set or admission profile.

USER-ADMISSION =
With USER-ADMISSION, the user ID is entered under which the modified admission profile is saved. FT requests which use this profile access the entered user ID in the local system. If the FTAC administrator has created an admission profile for a user, the user must enter ACCOUNT and PASSWORD in the operand USER-ADMISSION using MODIFY-FT-PROFILE (since he is the only person who should know them), before the profile can be used (see CREATE-FT-PROFILE, page 127).

USER-ADMISSION = *UNCHANGED
The USER-ADMISSION of this admission profile remains unchanged.

USER-ADMISSION = *OWN
For USER-IDENTIFICATION and ACCOUNT, the specifications for your user ID and your account number are taken from your LOGON authorization. A BS2000 password is only taken from your LOGON authorization when an FT request accesses the admission profile.

USER-ADMISSION = *PARAMETERS(...)
You can also enter the individual components of the user ID. This allows you to keep FT requests which use this admission profile under a different account number, for example. Or, a password can be set in the admission profile. FT requests which use this admission profile will then only function if their current LOGON password corresponds to the pre-set password.
The FTAC administrator can use these specifications to modify other users’ admission profiles.

USER-IDENTIFICATION =
With USER-IDENTIFICATION, you identify your user ID in BS2000.

USER-IDENTIFICATION = *OWN
Your user ID is taken from your LOGON authorization.

USER-IDENTIFICATION = <name 1..8>
This is the user ID with which the profile is to be associated.

ACCOUNT =
With ACCOUNT, you enter the account number under which an FT request is to be kept when it uses this admission profile.

ACCOUNT = *OWN
The account number is taken from your LOGON authorization.

ACCOUNT = *FIRST
The first account number assigned to the home pubset of the specified USER-IDENTIFICATION at the time the profile is used in the system. Used for account assignment in the case of transfer requests. If the ID's account number changes, the profile no longer has to be modified as was previously the case.

ACCOUNT = *NOT-SPECIFIED
The account number is to be specified by the owner of the admission profile. This function permits the FTAC administrator to set up profiles for user IDs whose account numbers he doesn't know.

ACCOUNT = <alphanum-name 1..8>
An FT request should be kept under the account number specified when it accesses this admission profile. You can enter any account number which is associated with your user ID.

PASSWORD =
With PASSWORD, you enter the BS2000 password associated with your user ID.

PASSWORD = *OWN
When an FT request refers to this admission profile, FTAC uses the BS2000 password valid for your user ID at that moment. This prevents you from having to modify the admission profile if the BS2000 password is changed.

PASSWORD = *NOT-SPECIFIED
The password is specified by the owner of the admission profile. This function permits the FTAC administrator to set up profiles for unknown user IDs

PASSWORD = *NONE
No BS2000 password is required for the user ID.

PASSWORD = <c-string 1..8> / <c-string 9..32> / <x-string 1..16>
When an FT request accesses the admission profile, the specified password is compared with the current LOGON password. If the two do not correspond, the FT request is rejected.

PASSWORD = *SECRET
The system prompts you to enter the password. However, this does not appear on the screen.

INITIATOR =
With INITIATOR, you determine if initiators from local and/or remote systems are permitted to use this admission profile for their FT requests.

INITIATOR = *UNCHANGED
The settings in this admission profile remain unchanged.

INITIATOR = *REMOTE
This admission profile may only be used for FT requests by initiators from remote systems.

INITIATOR = *LOCAL
This admission profile may only be used for FT requests by initiators from the local system.

INITIATOR = (*LOCAL,*REMOTE)
This admission profile may be used by initiators from local and remote systems.

TRANSFER-DIRECTION =
With TRANSFER-DIRECTION, you determine which transfer direction may be used with this admission profile.

The transfer direction is always determined from the system in which the admission profile was defined.

TRANSFER-DIRECTION = *UNCHANGED
The specification in the admission profile remains unchanged.

TRANSFER-DIRECTION = *NOT-RESTRICTED
With this admission profile, files can be transferred to and from a partner system.

TRANSFER-DIRECTION = *FROM-PARTNER
With this admission profile, files can only be transferred from a partner system to your system. It is not possible to display file attributes/directories (partial components of “inbound file management”).

TRANSFER-DIRECTION = *TO-PARTNER
With this admission profile, files can only be transferred from your system to a partner system. It is not possible to modify file attributes or delete files (partial components of “inbound file management”).

PARTNER =
With PARTNER, you can specify that this admission profile is to be used only for FT requests which are processed by a a certain partner system.

PARTNER = *UNCHANGED
Any partner in the admission profile remains unchanged.

PARTNER = *NOT-RESTRICTED
This admission profile's scope of use is not limited to FT requests with certain partner systems.

PARTNER = *ADD(NAME = list-poss(50): <text 1..200 with-low>)
With this specification, you can add elements to an existing list of partner systems. A maximum of 50 partner systems can be specified.

PARTNER = *REMOVE(NAME = list-poss(50): <text 1..200 with-low>)
With this specification, you can remove elements from an existing list of partner systems. A maximum of 50 partner systems can be specified.

PARTNER = list-poss(50): <text 1..200 with-low>
The admission profile only permits those FT requests which are processed with the specified partner systems. A maximum of 50 partner systems can be specified.

For PARTNER you can specify the name from the partner list or the fully qualified address of the partner system, see also section "Specifying partner addresses" on page 39. You are advised to use the name from the partner list.

MAX-PARTNER-LEVEL =
With MAX-PARTNER-LEVEL, a maximum security level can be specified. The admission profile will then only permit those FT requests which are processed with partner systems which have this security level or lower.
MAX-PARTNER-LEVEL works in conjunction with the admission set. When non-privileged admission profiles are used, the access check is executed on the basis of the smallest specified value.

MAX-PARTNER-LEVEL = *UNCHANGED
The specification for MAX-PARTNER-LEVEL in this admission set remains unchanged.

MAX-PARTNER-LEVEL = *NOT-RESTRICTED
If FT requests are processed with this admission profile, then the highest accessible security level is determined by the admission set.

MAX-PARTNER-LEVEL = <integer 0..100>
With this admission profile, all partner systems which have this security level or lower can be communicated with.

When you set MAX-PARTNER-LEVEL=0, you prevent access to the admission profile (for the time being). No FT request can then be processed with this admission profile.

FILE-NAME =
With FILE-NAME, you determine which files or library members under your user ID may be accessed by FT requests that use this admission profile.

FILE-NAME = *UNCHANGED
The specifications for FILE-NAME in this admission profile remain unchanged.

FILE-NAME = *NOT-RESTRICTED
The admission profile permits unrestricted access to all files and library members of the user ID.

FILE-NAME = <filename 1..54> / <c-string 1..512 with-low> /
***POSIX(NAME=<posix-pathname 1..219>)**
With this admission profile, only the specified file may be accessed.
However, *open*FT is also able to generate unique filenames automatically, thus providing an easy way of avoiding conflicts. This is done by specifying the string %UNIQUE at the end of the filename which is predefined here (see the section “File names” in the User Guide). When follow-up processing is specified, this file can be referenced with %FILENAME.
You can also directly specify file transfer with pre- and post-processing here by entering the pipe symbol ’|’ followed by a command.

FILE-NAME = *EXPANSION (PREFIX=<filename 1..53> / <partial-filename 2..53> /
<c-string 1..511 with-low>)
This entry can be used to restrict access to a number of files which all begin with the same prefix. If a filename is entered in an FT request which uses this admission profile, FTAC sets the prefix defined with EXPANSION in front of this filename. The FT request is then permitted to access the file *PrefixFilename*.

Example
PREFIX=DONALD.; An FT request in which the FILE-NAME=DUCK is specified accesses the file DONALD.DUCK.

Please note that the part of a DVS filename which is specified in the file transfer command still has to be of the type <filename>.
If you want to perform file transfer with pre- or post-processing, you should indicate this by entering the pipe symbol ‘|’ at the start of the prefix. The created FTAC profile can then be used only for file transfer with pre- or post-processing since the file name that is generated also starts with a ‘|’. The maximum length of the full pre- or post-processing command is restricted to the maximum filename length.

Example
```
FILE-NAME=C‘|/Command1;/Command2;/Command3; ...‘
```

In this case, the characters available for the pre- or post-processing command in the FT request are limited to:
– alphanumeric characters (letters and digits)
– the special characters + = / ! - , @ ␣ , _ ” $ ’
– a period (.) between alphanumeric characters

In the case of admission profiles which are to be used exclusively for the ftexec command you must specify a filename or filename prefix that starts with the character string '|ftexecsv' (see CREATE-FT-PROFILE example 2).

FILE-NAME = *LIBRARY-ELEMENT(...)
With *LIBRARY-ELEMENT, you determine which of your libraries and library members may be accessed by FT requests which use this admission profile.

LIBRARY =
With LIBRARY, you define which libraries may be accessed with this admission profile.

LIBRARY = *UNCHANGED
The library specifications in the admission profile remain unchanged.

LIBRARY = *NOT-RESTRICTED
The admission profile does not restrict access to libraries.

LIBRARY = <filename 1..54>
With this admission profile, only this library may be accessed.

LIBRARY = *EXPANSION (PREFIX=<filename1..63 with-under> / <partial-filename 2..63>)
With this admission profile, only those libraries may be accessed which begin with the specified prefix. FTAC sets the prefix in front of a library name in an FT request which uses this admission profile, and then permits access to the library *Prefix-Libraryname.*

ELEMENT =
With ELEMENT, you determine which library members may be accessed with this admission profile.

ELEMENT = *UNCHANGED
The library member specifications in the admission profile remain unchanged.

ELEMENT = *NOT-RESTRICTED
The admission profile permits unrestricted access to library members.

ELEMENT = <composed-name 1..64 with-under>(...)
The admission profile only permits access to the specified library member.

VERSION =
With VERSION, access is only permitted for a specific version of the library member.

VERSION = *STD
The admission element permits access only to the highest version of the library member.

VERSION = <text 1..24>
Access is only permitted for this version of the library member.

ELEMENT = *EXPANSION (PREFIX=<partial-filename 2..63> /
<composed-name 1..63 with-under>)

With *EXPANSION, you define a prefix. When a name for a library member is specified in an FT request which uses this admission profile, FTAC adds the specified prefix to this member name. The admission profile then permits access to this member with the name *PrefixMembername*.

TYPE =

With TYPE, you can specify a certain type of library member. The admission profile then only permits access to library members of this type.

TYPE = *UNCHANGED

Any access restrictions to individual member types remain unchanged.

TYPE = *NOT-RESTRICTED

Access is not restricted to a certain type of library member.

TYPE = <name 1..8>

FT requests which use this admission profile may only access library members of this type.

FILE-PASSWORD =

With FILE-PASSWORD, you can enter a password for files into the admission profile. The FTAC functionality then only permits access to files which are protected with this password and to unprotected files. When a FILE-PASSWORD is specified in an admission profile, the password may no longer be specified in an FT request which uses this admission profile. This allows you to permit access to certain files to users in remote systems, without having to disclose the file passwords.

FILE-PASSWORD = *UNCHANGED

The specifications for FILE-PASSWORD in this admission profile remain unchanged.

FILE-PASSWORD = *NOT-RESTRICTED

The admission profile permits access to all files. If a password is set for a file, then it must be specified in the transfer request.

FILE-PASSWORD = *NONE

The admission profile only permits access to files without file passwords.

FILE-PASSWORD = <c-string 1..4> / <x-string 1..8> /
<integer -2147483648..2147483647>

The admission profile only permits access to files which are protected with the password specified and to unprotected files. The password which has already been specified in the profile may not be repeated in the transfer request. PASSWORD=*NONE would be entered in this case!

FILE-PASSWORD = *SECRET

The system prompts you to enter the password. However, this does not appear on the screen.

PROCESSING-ADMISSION =
With PROCESSING-ADMISSION, you can enter a user ID in your BS2000 system. Any follow-up processing of an FT request will be executed under this user ID. With PROCESSING-ADMISSION in the admission profile, you do not need to disclose your LOGON authorization to partner systems for follow-up processing.

PROCESSING-ADMISSION = *UNCHANGED
The PROCESSING-ADMISSION in this admission profile remains unchanged.

PROCESSING-ADMISSION = *SAME
For the PROCESSING-ADMISSION, the values of the USER-ADMISSION are used. If *SAME is entered here, then any FT request which uses this profile must also contain PROCESSING-ADMISSION=*SAME or PROCESSING-ADMISSION= *NOT-SPECIFIED.

PROCESSING-ADMISSION = *NOT-RESTRICTED
FT requests which use this admission profile may contain any PROCESSING-ADMISSION. For follow-up processing with FTAM partners, PROCESSING-ADMISSSION must have a value not equal to *NOT-RESTRICTED.

PROCESSING-ADMISSION = *PARAMETERS(...)
You can also enter the individual components of the user ID. This allows you to keep FT requests which use this admission profile under a different account number, for example. Or, a password can be set in the admission profile. FT requests which use this admission profile will then only function if their current LOGON password corresponds to the pre-set password.

USER-IDENTIFICATION =
With USER-IDENTIFICATION, you enter the user ID under which the follow-up processing is to be executed.

USER-IDENTIFICATION = *SAME
The USER-IDENTIFICATION is taken from the USER-ADMISSION.

USER-IDENTIFICATION = *NOT-RESTRICTED
The admission profile does not restrict the user ID for the follow-up processing.

USER-IDENTIFICATION = <name 1..8>
FT requests which are processed with this admission profile are only permitted follow-up processing under this user ID. If another user ID is entered here, the parameter PASSWORD must also be entered. PASSWORD=*SAME is then not valid.

ACCOUNT =
With ACCOUNT, you specify the account number for the follow-up processing.

ACCOUNT = *SAME
The account number is taken from the USER-ADMISSION.

ACCOUNT = *NOT-RESTRICTED
The account number may be specified in FT requests that work with the admission profile. The admission profile does not restrict the account for follow-up processing.

ACCOUNT = <alphanum-name 1..8>
Follow-up processing is to be settled under this account number.

PASSWORD =
With PASSWORD, you specify, where applicable, the BS2000 password for the user ID under which the follow-up processing is to be executed. Here, you can enter a PASSWORD when the user ID in question doesn't have such a password (yet).

PASSWORD = *SAME
The value *SAME is only valid if the PROCESSING-ADMISSION refers to your own user ID. If PASSWORD=*OWN is entered on USER-ADMISSION, then the BS2000 password valid at the time of the request is used for the PROCESSING-ADMISSION. The entry *SAME is only possible here if the follow-up processing is not started with the command /ENTER.

PASSWORD = *NOT-RESTRICTED
The password may be specified for FT requests which work with the admission profile. The admission profile does not restrict the password for follow-up processing.

PASSWORD = *NONE
FT requests which use this admission profile can only initiate follow-up processing on user IDs without a password.

PASSWORD = <c-string 1..8> / <c-string 9..32> / <x-string 1..16>
FT requests which use the admission profile may only initiate follow-up processing on user IDs which are protected with this BS2000 password.

PASSWORD = *SECRET
The system prompts you to enter the password. The entry does not appear on the screen.

SUCCESS-PROCESSING =
With SUCCESS-PROCESSING, you can restrict the follow-up processing which an FT request is permitted to initiate in your system after a successful data transfer.

SUCCESS-PROCESSING = *UNCHANGED
The specifications for SUCCESS-PROCESSING in this admission profile remain unchanged.

SUCCESS-PROCESSING = *NOT-RESTRICTED
In FT requests which use this admission profile the operand SUCCESS-PROCESSING may be used without restriction.

SUCCESS-PROCESSING = *NONE
The admission profile does not permit follow-up processing after successful data transfer.

SUCCESS-PROCESSING = <c-string 1..1000 with-low>
Specifies the BS2000 commands which are executed in the local system after successful data transfer. Individual commands must be preceded by a slash (/). The individual commands must be separated by a semicolon (;). If a character string is enclosed by single or double quotes (’ or ”) within a command sequence, *open*FT (BS2000) does not interpret any semicolons within this character string as a separator.

SUCCESS-PROCESSING = *EXPANSION(...)
If a SUCCESS-PROCESSING was specified in an FT request which uses this admission profile, FTAC adds the prefix or suffix specified here to this command. As follow-up processing, the command which has been thus expanded is then executed.

PREFIX = *UNCHANGED
The specifications for the follow-up processing prefix in this admission profile remain unchanged.

PREFIX = *NOT-RESTRICTED
Follow-up processing is not restricted by a prefix.

PREFIX = <c-string 1..999 with-low>
The specified prefix is set in front of a command which is specified in an FT request as follow-up processing. Then, the command which has been expanded with the prefix is executed as follow-up processing.

SUFFIX = *UNCHANGED
The specifications for the follow-up processing suffix in this admission profile remain unchanged.

SUFFIX = *NOT-RESTRICTED
Follow-up processing is not restricted by a suffix.

SUFFIX = <c-string 1..999 with-low>
The specified prefix is set after a command which is specified in an FT request as follow-up processing. Then, the command which has been expanded with the suffix is executed as follow-up processing.

Example 1
If PREFIX='/PRINT-FILE␣' is defined and SUCC='filename' specified in the FT request, then FT executes the command “/PRINT-FILE␣filename” as follow-up processing.

Example 2
If SUFFIX='filename' is defined and SUCC='/PRINT-FILE' specified in the FT request, then FT executes the command “/PRINT-FILE␣filename” as follow-up processing.

If a suffix or prefix is defined at this point, then no command sequence for the follow-up processing may be specified in FT requests which use this admission profile. This makes the setting of prefixes and suffixes mandatory.

FAILURE-PROCESSING =
With FAILURE-PROCESSING, you can restrict the follow-up processing which an FT request is permitted to initiate in your system after a failed data transfer.

FAILURE-PROCESSING = *UNCHANGED
The specifications for FAILURE-PROCESSING in this admission profile remain unchanged.

FAILURE-PROCESSING = *NOT-RESTRICTED
In FT requests which use this admission profile the operand FAILURE-PROCESSING may be used without restriction.

FAILURE-PROCESSING = *NONE
The admission profile does not permit follow-up processing after failed data transfer.

FAILURE-PROCESSING = <c-string 1..1000 with-low>
Specifies the BS2000 commands which are executed in the local system after failed data transfer. Individual commands must be preceded by a slash (/). The individual commands must be separated by a semicolon (;). If a character string is enclosed by single or double quotes (' or ") within a command sequence, *open*FT (BS2000) does not interpret any semicolons within this character string as a separator.

FAILURE-PROCESSING = *EXPANSION(...)
If a FAILURE-PROCESSING was specified in an FT request which uses this admission profile, FTAC adds the prefix or suffix specified here to this command. As follow-up processing, the command which has been thus expanded is then executed.

If a suffix or prefix is defined at this point, then no command sequence for the follow-up processing may be specified in FT requests which use this admission profile. This makes the setting of prefixes and suffixes mandatory.

PREFIX = *UNCHANGED
The specifications for the follow-up processing prefix in this admission profile remain unchanged.

PREFIX = *NOT-RESTRICTED
Follow-up processing is not restricted by a prefix.

PREFIX = <c-string 1..999 with-low>
The specified prefix is set in front of a command which is specified in an FT request as follow-up processing. Then, the command which has been expanded with the prefix is executed as follow-up processing.

SUFFIX = *UNCHANGED
The specifications for the follow-up processing suffix in this admission profile remain unchanged.

SUFFIX = *NOT-RESTRICTED
Follow-up processing is not restricted by a suffix.

SUFFIX = <c-string 1..999 with-low>
The specified prefix is set after a command which is specified in an FT request as follow-up processing. Then, the command which has been expanded with the suffix is executed as follow-up processing.

WRITE-MODE =
With WRITE-MODE, you determine the WRITE-MODE which is valid for this FT request. WRITE MODE is only effective if the receive file is in the same system as the admission profile definition.

WRITE-MODE = *UNCHANGED
The specifications for WRITE-MODE in this admission profile remain unchanged.

WRITE-MODE = *NOT-RESTRICTED
In an FT request which accesses this admission profile, WRITE-MODE may be used without restrictions.

WRITE-MODE = *NEW-FILE
In the FT request, *NEW-FILE, *REPLACE-FILE or *EXTEND-FILE may be entered for WRITE-MODE. If the receive file already exists, the transfer will be rejected.

WRITE-MODE = *REPLACE-FILE
In the FT request, only *REPLACE-FILE or *EXTEND-FILE may be entered for WRITE-MODE.

WRITE-MODE = *EXTEND-FILE
In the FT request, only *EXTEND-FILE may be entered for WRITE-MODE.

FT-FUNCTION =
This operand permits the restriction of the profile validity to certain FT functions (=file transfer and file management functions).

FT-FUNCTION = *UNCHANGED
The existing scope of file management functions remains unchanged.

FT-FUNCTION = *NOT-RESTRICTED
The full scope of FT functions is available. For reasons of compatibility, the specification NOT-RESTRICTED means that FILE-PROCESSING is not permitted! All other functions are permitted if this value is specified.

FT-FUNCTION = (*TRANSFER-FILE,*MODIFY-FILE-ATTRIBUTES,
***READ-DIRECTORY,*FILE-PROCESSING)**
The following file transfer functions are available:

***TRANSFER-FILE**
The admission profile may be used for the file transfer functions “transfer files”, “view file attributes” and “delete files”.

***MODIFY-FILE-ATTRIBUTES**
The admission profile may be used for the file transfer functions “view file attributes” and “modify file attributes”.

***READ-DIRECTORY**
The admission profile may be used for the file transfer functions “view directories” and “view file attributes”.

***FILE-PROCESSING**
The admission profile may be used for the “preprocessing” and “postprocessing” file transfer functions. The “transfer files” function must also be permitted.

The *FILE-PROCESSING specification is of relevance only for FTAC profiles without a filename prefix. Otherwise the first character of the filename prefix determines whether only normal data transfer (no pipe symbol “|”) or only pre- and post-processing (pipe symbol “|”) are to be possible with this FTAC profile.

USER-INFORMATION =
Here the user of FTAC administrator can specify a text in the admission profile. This text can be displayed with the SHOW-FT-PROFILE command.

USER-INFORMATION = *UNCHANGED
Any existing text remains unchanged.

USER-INFORMATION = *NONE
Any existing text is deleted.

USER-INFORMATION = <c-string 1..100 with-low>
The character string entered is accepted as user information.

DATA-ENCRYPTION =
Using DATA-ENCRYPTION, you can specify whether user data with this profile must be transferred in encrypted form.

DATA-ENCRYPTION = *UNCHANGED
The encryption option should remain unchanged.

DATA-ENCRYPTION = *NOT-RESTRICTED
The encryption option for user data is not restricted. File transfer requests with encryption and file transfer requests without encryption are both accepted.

DATA-ENCRYPTION = *NO
Only file transfer requests that do not have encrypted user data are accepted, i.e. requests with encryption are rejected.
If the request is made in a BS2000, DATA-ENCRYPTION=*NO must be specified there in the NCOPY request.

DATA-ENCRYPTION = *YES

Only file transfer requests that have encrypted user data are accepted, i.e. requests without encryption are rejected.
If the request is made in a BS2000, for example, then DATA-ENCRYPTION=*YES must be specified there in the NCOPY request.

When using restrictions for FILE-NAME, SUCCESS-PROCESSING and FAILURE-PROCESSING, keep in mind that

- a restriction for follow-up processing must always be made for SUCCESS- and FAILURE-PROCESSING. Otherwise, it is possible that users will avoid this step.
- PREFIX of FILE-NAME, SUCCESS-PROCESSING and FAILURE-PROCESSING must correspond
  (e.g. FILE-NAME=*EXP(XYZ.),SUCC=*EXP('/PRINT-FILE XYZ.').

Example

After Donald Duck has created an admission profile with the name *profile1*, which permits other users access to his user ID with the LOGON authorization, he decides he wants to restrict this profile so that only FT accesses are possible to files which begin with the prefix *BRANCH*.

The required command is:

```
/MODIFY-FT-PROFILE␣NAME=profile1,
                   FILE-NAME=*EXPANSION(PREFIX=branch.)
```

A possible short form of this command is:

```
/MOD-FT-PROF␣profile1,FILE-N=(PRE=branch.)
```

This places heavy restrictions on the admission profile. The other specifications remain unchanged.

Command return codes

(SC2)	SC1	Maincode	Meaning
0	0	FTC0051	A user ID with the same name already exists in the system.
0	64	FTC0053	No FT profile exists which meets the criteria specified.
0	64	FTC0055	The partner restrictions were lifted.
0	0	FTC0056	Transfer admission is blocked.
0	64	FTC0100	An FT profile with this name already exists.
0	64	FTC0101	An FT profile with this transfer admission already exists.
0	64	FTC0150	The access password is missing.
0	64	FTC0151	Modifications can only be made by the administrator or owner.
0	64	FTC0153	The owner ID entered is not the user's own ID.
0	64	FTC0170	The partner entered is unknown within the partner system available for this user.
0	64	FTC0171	The profile entered does not exist.
0	64	FTC0172	The user admission entered does not exist in the system.
0	64	FTC0173	The processing admission entered does not exist in the system.
0	64	FTC0174	The parameters "NEW-NAME" and "TRANSFER-ADMISSION" may only used together in conjunction with unique selection criteria ("NAME" or "TRANSFER-ADMISSION").
0	64	FTC0178	The partner name entered occurs several times.
0	64	FTC0179	The maximum number of partner restrictions has been exceeded.
0	64	FTC0182	The maximum length of partner names has been exceeded.
0	64	FTC0200	The total length of the two follow-up processing commands is too long.
0	64	FTC0255	A system error has occurred.

SC1/2 = Subcode 1/2 in decimal notation

For additional information, see section "Command return codes" on page 102

4.24 REMOVE-FT-PARTNER (FTREMPTN) Remove remote system from partner list

The command REMOVE-FT-PARTNER (or FTREMPTN) is used to remove a remote system from the partner list of the local system.

If a partner system is deleted from the partner list then all requests involving this partner system are aborted. REMOVE-FT-PARTNER therefore represents a simple way to delete all the requests relating to a given partner. A request to a partner removed with FTREMPTN is eliminated even if the request is already known in the partner system (in the same way as with CANCEL-FILE-TRANSFER .. FORCE-CANCELLATION=*YES).

REMOVE-**FT-PART**NER / **FTREMPTN**
PARTNER = <text 1..200 with-low>

Operands

PARTNER = <text 1..200 with-low>
Specifies the name of the partner system from the partner list or the address of the partner system.

Command return codes

(SC2)	SC1	Maincode	Meaning
83	32	CMD0221	Internal error.
35	64	FTR1035	User not authorized for this command.
45	64	FTR1045	Partner name not found in partner list.
1	0	FTR1048	Active requests could not yet be deleted.

SC1/2 = Subcode 1/2 in decimal notation
For additional information, see section “Command return codes” on page 102

Example

Remove the remote system PARTNER1 from the partner list of the local system:

```
/REMOVE-FT-PARTNER PARTNER=PARTNER1
```

4.25 SHOW-FILE-TRANSFER or SHFT/NSTATUS/FTSHWREQ Query status of FT request

The SHOW-FILE-TRANSFER (or SHFT or NSTATUS or FTSHWREQ) command allows the FT user to request information about FT requests. As with CANCEL-FILE-TRANSFER, the user can specify selection criteria in order to obtain information about specific FT requests. The FT administrator can obtain information about the requests of any owner. An FT user can, however, only request details of requests of which he or she is the owner.

The owner of requests issued in the local system is the user ID under which they are submitted.

The owner of requests issued in the remote system is the user ID in the local system under which the requests are executed.

The scope of information to be output can be selected. By default the following information is output by the system in response to the SHOW-FILE-TRANSFER command:

– the transfer ID of the request,
– the initiator of the request (local or remote system),
– the operating status of the request (see description of operands for more details),
– the partner,
– the transfer direction,
– the name of the file (or library member) to be transferred in the local system.

By entering INFORMATION=*ALL in the SHOW-FILE-TRANSFER command more information can be obtained. *open*FT then, in addition to the standard output, gives the values of the following operands of the request (operands of the TRANSFER-FILE command used to issue the request):

– PRIO
  With which priority is the request to be executed?
– COMPRESS
  Is the data to be transferred in compressed form?
– WRITE
  How is the receive file to be written?
– START
  What is the earliest start time for the request?
– CANCEL
  When is the request to be canceled?

– OWNER
  Who is the owner of the request?
– DATA
  How is the data to be transferred (binary/character)?
– TRANSP
  Should the data be transferred in transparent file format?
– ENCRYPT
  Should the user data be transferred in encrypted form?
– DICHECK
  Is data integrity checking in use?
– TRANSFER-ADMISSION (local)
  Which transfer admission for the local system was specified in the FT request?
– PROCESSING-ADMISSION (local)
  Which authorization for the follow-up processing in the local system was specified in the FT request?
– PROCESSING (local)
  Which commands were specified in the FT request for follow-up processing in the local system (SUCCESS/FAILURE)?
  The follow-up processing data is only output if the issuer of the command owns the request. Otherwise, *SECRET is output here.

The following additional information is given for requests issued locally:

– LISTING
  Where is the result list to be sent?
– ASYN-MSG
  Specifies whether and which request result leads to an asynchronous termination message.
  Possible values: ALL, FAIL.
– TRANSFER-ADMISSION (remote)
  Which transfer admission for the remote system was specified in the FT request?
– PROCESSING-ADMISSION (remote)
  Which authorization for follow-up processing in the remote system was specified in the FT request?
– PROCESSING (remote)
  Which commands were specified in the FT request for follow-up processing in the remote system (SUCCESS/FAILURE)?

  The follow-up processing data is only output if the issuer of the command owns the request. Otherwise, in this case, *SECRET is output.

– CCSN (local and remote)
  CCS name used in the local and/or remote system when reading the file.

See the examples at the end of this section for more details on how these operands appear in the command output.

The more precise your information request, the fewer irrelevant requests are output.

SHOW-FILE-TRANSFER / SHFT / NSTATUS / FTSHWREQ

```
TRANSFER-ID = *ALL / <integer 1..2147483639>
,SELECT = *OWN / *PARAMETERS(...)
*PARAMETERS(...)
    OWNER-IDENTIFICATION = *OWN / *ALL / <name 1..8>
    ,INITIATOR = (*LOCAL, *REMOTE) / list-poss(2): *LOCAL / *REMOTE
    ,PARTNER = *ALL(...) / <text 1..200 with-low>
        *ALL(...)
            PARTNER-STATE = *ALL / *ACTIVE
    ,FILE = *ALL / <filename 1..54> / <c-string 1..512 with-low> / *LIBRARY-ELEMENT(...) /
                *POSIX(NAME=<posix-pathname 1..219>) / *PUBSET(PUBSET=<cat-id 1..4>)
        *LIBRARY-ELEMENT(...)
            LIBRARY = *ALL / <filename 1..54>
            ,ELEMENT = *ALL / <filename 1..64 without-gen-vers>(...) /
                              <composed-name 1..64 with-under>(...)
                <filename>(...) / composed-name>(...)
                    VERSION = *ALL / <text 1..24>
            ,TYPE = *ALL / <name 1..8>
    ,MONJV = *NONE / <filename 1..54 without-gen-vers>
    ,JV-PASSWORD = *NONE / <c-string 1..4> / <x-string 1..8> /
                       <integer -2147483648..2147483647> / *SECRET
    ,STATE = *ALL / *SUSPEND / *LOCKED / *WAIT / *ACTIVE / *FINISHED / *HOLD
,INFORMATION = *STD / *ALL / *SUMMARY
,OUTPUT = *SYSOUT(...) / *SYSLST(...)
    *SYSOUT(...) / *SYSLST(...)
        LAYOUT = *STD / *CSV
```

Operands

TRANSFER-ID =
Transfer ID of the FT request about which information is required.

TRANSFER-ID = *ALL
supplies information about all the owner's FT requests. You can obtain information about all current FT requests that access your system.

TRANSFER-ID = <integer 1..2147483639>
is the transfer ID assigned to the local system and output as part of the message confirming acceptance of the request.

SELECT =
contains selection criteria defining the file transfer requests on which inquiries are to be made. Information on a file transfer request is output if the request satisfies all the specified criteria.

SELECT = *OWN
provides information on all current file transfer requests for which you are designated as the owner.

SELECT = *PARAMETERS(...)

OWNER-IDENTIFICATION =
Owner of the FT request. Only the FT administrator can make use of this operand unrestricted; other users may only enter their own user ID.

OWNER-IDENTIFICATION = *OWN
provides information only on the file transfer requests in the user's own ID.

OWNER-IDENTIFICATION = *ALL
provides information on FT requests in all user IDs.

OWNER-IDENTIFICATION = <name 1..8>
identifies a specific user ID about whose file transfer requests information is required.

INITIATOR =
defines the initiator of the file transfer requests concerned.

INITIATOR = (*LOCAL,*REMOTE)
provides information on file transfer requests in the local system and in remote systems.

INITIATOR = *LOCAL
provides information on file transfer requests issued in the local system.

INITIATOR = *REMOTE
provides information on file transfer requests issued in the remote systems.

PARTNER =
selects file transfer requests carried out with a specified remote system.

PARTNER = *ALL(...)
The partner system is not used as a selection criterion to determine the file transfer requests on which information is to be output.

PARTNER-STATE =
The status of the partner system is used as a selection criterion.

PARTNER-STATE = *ALL
The requests are selected independently of the partner system's status.

PARTNER-STATE = *ACTIVE
Only the requests to and from the active partners are selected.

PARTNER = <text 1..200 with-low>
is the name or an address of a partner system. Information is required on the file transfer requests being executed with this system. For more information on address specifications, see section "Specifying partner addresses" on page 39.

FILE =
selects the FT requests that access this file, this pubset or this library member in the local system as a send file or receive file. The file name or library member name must be specified exactly as it appears in the FT request. If %UNIQUE was specified, the file name generated by *open*FT must be entered as the selection criterion here.

FILE = *ALL
The file name is not used as a selection criterion to define the file transfer requests on which information is to be output.

FILE = <filename 1..54> / <c-string 1..512 with-low>
***POSIX(NAME = <posix-pathname 1..219>)**
is the name of a file. Information is required on the file transfer requests that access this file.

FILE = *PUBSET(PUBSET = <cat-id 1..4>)
Information on all FT requests that have locked files on the specified pubset should be displayed.

FILE = *LIBRARY-ELEMENT(...)
Information is required on file transfer requests that access library members in the local system.

LIBRARY =
selects the library concerned.

LIBRARY = *ALL
The library name is not used as a selection criterion to define the file transfer requests on which information is to be output.

LIBRARY = <filename 1..54>
specifies the name of a library. Information is required on the file transfer requests that access this library.

ELEMENT =
selects a library member. Information is required on all the file transfer requests that access this member.

ELEMENT = *ALL
The name of the library member is not used as a selection criterion to define the file transfer requests.

ELEMENT = <filename 1..64 without-gen-vers>(...)
<composed-name 1..64 with-under>(...)
is the name of a library member. Information is required on the file transfer requests that access this library member.

VERSION =
specifies the version number of the library member.

VERSION = *ALL
Information is required on all file transfer requests that access any version of the library member.

VERSION = <text 1..24>
Information is required on the file transfer requests that access a specific version of the library member.

TYPE =
The type of library member.

TYPE = *ALL
The member type is not used as a selection criterion to define the file transfer requests on which information is to be output.

TYPE = <name 1..8>
Information is required only on those file transfer requests that access library members of this type.

MONJV =
If appropriate, selects the specific file transfer request that is being monitored by this job variable.

MONJV = *NONE
A job variable is not used as a selection criterion to define the file transfer request on which information is to be output.

MONJV = <filename 1..54 without-gen-vers>
Information is required on the file transfer request that is being monitored by this job variable.

JV-PASSWORD =
If required, specifies the password needed to access the job variable.
If you have already notified the system of the password with the BS2000 command ADD-PASSWORD, you do not have to specify JV-PASSWORD.

JV-PASSWORD = *NONE
The job variable is not password-protected.

JV-PASSWORD = <c-string 1..4> / <x-string 1..8> /
<integer -2147483648..2147483647>
This password is required for the job variable.

JV-PASSWORD = *SECRET
The system requests you to enter the password. This input is not displayed on the screen.

STATE =
selects those file transfer requests that are in the specified status. The status of a request may change in between entry of the command and information output. This is why the output may include requests that are in a state other than the one selected with STATE.

STATE = *ALL
The status of a request is not used as a selection criterion to define the file transfer requests on which information is to be output.

STATE = *SUSPEND
requests information on those file transfer requests that are currently in SUSPEND status (= interrupted).

STATE = *LOCKED
requests information on those file transfer requests that are currently in LOCKED status (= temporarily locked as a result of a longer term resource bottleneck).

STATE = *WAIT
requests information on those file transfer requests that are currently in WAIT status (= waiting for resources).

STATE = *ACTIVE
requests information on those file transfer requests that are currently in ACTIVE status (= being processed).

STATE = *FINISHED
requests information on those file transfer requests that are currently in FINISHED status (= terminated or aborted, but where the user has not yet been informed).

STATE = *HOLD
requests information on those FT requests that are currently in HOLD status (= awaiting the specified start time).

INFORMATION =
Scope of the output.

INFORMATION = *STD
Output is summary form and contains the following information:
– Transfer ID,
– Status,
– Initiator,
– State,
– Partner,
– Direction of transfer,
– Byte count,
– File or library member name in the local system.
(see example 1).

INFORMATION = *ALL
Output is in full form. In addition to the summary form data, further information is provided on the operands used in the TRANSFER-FILE(NCOPY) command (see example 2).

INFORMATION = *SUMMARY
Output is in the form of a specified sum. By specifying INFORMATION=*SUMMARY, you can restrict the output information to a statistic of the currently existing requests. By doing this, the display is arranged according to the conditions in which the requests find themselves. The displayed sum can, of course, exceed the sum of the individual columns, since all requests, even those that still have no request condition, are counted.

OUTPUT =
Defines the output medium.

OUTPUT = *SYSOUT(...)
The output is sent to SYSOUT.

OUTPUT = *SYSLST(...)
The output is sent to SYSLST.

LAYOUT = *STD
The output is formatted using a standard layout that can be easily read by the user.

LAYOUT = *CSV
The output is supplied in CSV (Comma Separated Value) format. This is a widely used tabular format, especially in the PC environment, in which individual fields are separated by a delimiter, which is usually a semicolon ";" (see page 104 and 294).

The specification of more than one selection criterion in the SHOW-FILE-TRANSFER command may result in a file transfer request being “overdefined” (e.g. by entries for TRANSFER-ID and MONJV).
If, in such a case, the specified criteria conflict, the command is acknowledged with the following message:

```
% FTR0504 No requests available for the selection criteria
```

In such a case, procedures do not branch to the next SET-JOB-STEP.

Command return codes

(SC2)	SC1	Maincode	Meaning
0	0	CMD0001	There are no requests that meet the specified selection criteria.
33	32	CMD0221	Request rejected. Internal error.
36	32	CMD0221	Request rejected. Request data inconsistent.
82	32	CMD0221	Internal error. Job variable not accessible.
83	32	CMD0221	Internal error.
88	32	CMD0221	Error during OPS generation.
36	64	FTR1036	User not authorized for other user IDs
47	64	FTR1047	The request with the specified transfer ID could not be found.
226	64	FTR2226	Job variable contents inconsistent.
227	64	FTR2227	Job variable not in use by *open*FT.
228	64	FTR2228	Job variable not found.

SC1/2 = Subcode 1/2 in decimal notation
For additional information, see section “Command return codes” on page 102

OPS variables

The following table shows the OPS variables for the command SHOW-FILE-TRANSFER with the operand INF=*ALL, the underlined values are valid for the output with the operand INF=*STD. The table on page 235 shows the OPS variables for the output with the operand INF=*SUMMARY.

Element	Type	Output
TRANS-ID	Integer	
STA	String	*SUSPEND / *LOCK / *WAIT / *ACTIVE / *FINISH / *HOLD
BYTE-COUNT	Integer	
PRIO	String	*NORM / *HIGH / *LOW

Element	Type	Output
INIT	String	*LOC / *REM
TRANS-DIRECT	String	*TO-PARTNER / *FROM-PARTNER
PARTNER-NAME	String	
COMPRESS	String	*NONE / *BYTE-REPETITION / *ZIP
DATA-ENC	String	*YES / *NO
DICHECK	String	*YES / *NO
WRITE-MODE	String	* REPL-FILE / *NEW-FILE / *EXT-FILE
FILE-SIZE	String	value
REC-SIZE	String	value
REC-FORMAT	String	*STD / *VARIABLE / *FIXED /*UNDEFINED
START	Struct	
.DATE	String	*SOON / yyyy-mm-dd
.TIME	String	*SOON / hh:mm:ss
CANCEL	Struct	
.DATE	String	*NO / yyyy-mm-dd
.TIME	String	*NO / hh:mm:ss
OWNER	String	
DATA-TYPE	String	
TRANSP	String	*YES / *NO
LOC-PAR	Struct	
.F-TYPE [1]	String	*FILE / *LIB
.F-NAME	String	
.LIB	String	
.ELEM	String	
.VERSION	String	
.TYPE	String	
.TRANS-ADMIS	Struct	
.USER-ID	String	
.ACCOUNT	String	
.PROF-NAME [2]	String	
.PROCESS-ADMIS	Struct	
.USER-ID	String	
.ACCOUNT	String	

Element	Type	Output
.SUCC-PROCESS	String	*SECRET / success-processing
.FAIL-PROCESS	String	*SECRET / failure-processing
.LISTING	String	*NONE / *SYSLST / *LISTFILE / *FAIL-SYSLST / *FAIL-LISTFILE
.MONJV	String	
.CCS-NAME	String	*STD / value
REM-PAR	Struct	
.F-TYPE [1]	String	*FILE / *LIB
.F-NAME	String	
.LIB	String	
.ELEM	String	
.VERSION	String	
.TYPE	String	
.TRANS-ADMIS	Struct	
.USER-ID [3]	String	*REM-PROF / user-id
.ACCOUNT [3]	String	*REM-PROF / account
.PROCESS-ADMIS	Struct	
.USER-ID	String	
.ACCOUNT	String	
.SUCC-PROCESS	String	*SECRET / success-processing
.FAIL-PROCESS	String	*SECRET / failure-processing
.CCS-NAME	String	*STD / value

[1] For F-Type=*FILE, LIB, ELEM, VERSION and TYPE are not displayed.

[2] USER-ID and ACCOUNT are not assigned if an FTAC profile is specified.

[3] Since this cannot be output when a remote FTAC transfer admission is specified, USER-ID and ACCOUNT are assigned with *REM-PROFILE in this case.

The following table shows the OPS variables for the output with the operand INF=*SUMMARY.

Element	Type	Output
NUM-ACTIVE	Integer	
NUM-WAIT	Integer	
NUM-LOCK	Integer	
NUM-SUSPEND	Integer	
NUM-HOLD	Integer	
NUM-FINISHED	Integer	
NUM-SUMM [1]	Integer	

[1] Sum of all requests including the requests that are still not validated and therefore not counted in any of the other elements.

Example **1**

Information is to be output to SYSOUT on those FT requests submitted by the remote system ALFRED which require access to the file DRAISINE and are currently active. The required command is as follows:

```
/SHOW-FILE-TRANSFER                                                    -
/                  SELECT=(INITIATOR=*REMOTE,                          -
/                  PARTNER=ALFRED,                                     -
/                  FILE-NAME=DRAISINE,                                 -
/                  STATE=*ACTIVE)
```

The recommended short form of this command is as follows:

```
/SHOW-FILE-TRANS SEL=(INIT=*REM,PART-NAME=ALFRED,
/     FILE-NAME=DRAISINE,STATE=*ACT
```

or

```
/NSTATUS SEL=(INIT=*REM,PART-NAME=ALFRED,FILE-NAME=DRAISINE,STATE=*ACT)
```

The information is then output in the following format, for example:

```
%TRANS-ID   INI STATE PARTNER  DIR  BYTE-COUNT  FILE-NAME
%528184     REM ACT   ALFRED   TO   14760       DRAISINE
```

The information is output to SYSOUT, since this is the default value for the output of inquiry information. Description of the output columns:

`TRANS-ID:` Transfer ID of the file transfer request

`INI:` Initiator of the file transfer request (REM for REMOTE, LOC for LOCAL)

`STATE:` State of the request (here ACT for ACTIVE, other outputs:

`SUSP` for SUSPEND,

`LOCK` for LOCKED,

`WAIT` for WAIT,

`FIN` for FINISHED,

`HOLD` for HOLD

`PARTNER:` Symbolic name of the relevant partner system.

If the FT request is in the STATE=WAIT state, and there is no normal internal resource bottleneck, then the partner name is preceded by one of the following characters:

* The FT administrator of the local system has locked a resource.

! An attempt to set up a connection to the partner system failed (possibly because the remote system is not running, for example, or because FT has not been started there or, in the case of TCP/IP connections, because the port specification contains *BY-Transportsystem and there is no BCMAP). This can also occur, if *open*FT has discovered an error during the internal check of transferred data integrity.

? Installation error. The PORT in BCMAP does not correspond to that in the partner entry. Check the installation. This can also occur if the authentication of the local or remote system has failed due to an unsuitable public key.

`DIR:` Transfer direction

`BYTE-COUNT:` Number of bytes transferred up to the last restart point (in the case of data compression this is the a number of bytes of compressed data)

`FILE-NAME:` Name of the relevant file or library member in the local system.

Example 2

Full information is to be output to SYSLST on the FT request with transfer ID 441121. If the file transfer request was issued under the same user ID as that under which the inquiry is made, then the command is as follows:

```
/SHOW-FILE-TRANSFER                                                    -
/                  TRANSFER-ID=441121,                                 -
/                  INFORMATION=*ALL,                                   -
/                  OUTPUT=*SYSLST
```

The recommended short form of this command is as follows:

```
/SHOW-FILE-TRANS 441121,INF=*ALL,OUTPUT=*SYSLST
```

The information output on SYSLST then has the following format, for example:

```
%TRANSFER-ID =330162      STORE  =06-12-04 17:03:03  FILESIZE=2048
%   STATE    =WAIT        BYTECNT=0
%   INITIATOR=LOCAL       TRANS  =TO                 PRIO    =NORM
%   WRITE    =REPLACE     START  =SOON               CANCEL  =NO
%   COMPRESS =NONE        DATA   =CHAR
%   TRANSP   =NO          ENCRYPT=NO
%   OWNER    =TSOS        DICHECK=NO
%   PARTNER  =HUGO
%   PARTNER-STATE = NOCON
%   LOC: FILE     =BULLETIN
%        TRANS-ADM=(USER,ACCOUNT)
%        ASYN-MSG =ALL
%        CCSN     =EDF041
%   REM: FILE     =TESTFILE
%        TRANS-ADM=REMOTE-PROFILE
%        CCSN     =UTF8
```

Description of the output fields:

`TRANSFER-ID:`
: Transfer ID of the request

`STATE:`
: State of the request

`BYTECNT:`
: Number of bytes transferred up to the last restart form (in the case of data compression in compressed form)

`PRIO:`
: Priority with which the request is to be started;
here: NORM for NORMAL.

`INITIATOR:`
: Initiator of the request

`TRANS:`
: Transfer direction, as seen from local system

PARTNER: Symbolic name of partner system participating in the request.
If the FT request is in the STATE=WAIT state, and there is no normal internal resource bottleneck, then the partner name is preceded by one of the following characters:

- * The FT administrator of the local system has locked a resource.
- ! An attempt to set up a connection to the partner system failed (possibly because the remote system is not running, for example, or because FT has not been started there or, in the case of TCP/IP connections, because the port specification contains *BY-Transportsystem and there is no BCMAP). This can also occur if the *open*FT has discovered an error while internally checking the integrity of the transferred data.
- ? Installation error. The PORT in BCMAP does not correspond to that in the partner entry. Check the installation. This can also occur if authentication of the local or remote system has failed due to an unsuitable public key.

COMPRESS: Indicates whether file transfer is in compressed form

WRITE: Indicates whether the receive file is overwritten or extended.

START: Requested start time of the request
(SOON for “as soon as possible”)

CANCEL: Requested abortion time
(NO for “no abortion requested”)

OWNER: Owner of request in local system

DATA: Type of file:

- CHAR for text file
- BIN for binary file
- NOT-SPECIFIED
  in TRANSFER-FILE (NCOPY), no DATA-TYPE was specified
- USER for user format

TRANSP: Specifies whether the transfer is to be done in transparent format

DICHECK: Specifies whether data integrity is to be checked (YES) or not (NO)

ENCRYPT: Specifies whether the file content is to be transferred in encrypted form

LOC: Entry for LOCAL-PARAMETER.

The entry can include more than in this example; the keywords correspond to the recommended abbreviations of the TRANSFER-FILE keywords; the meaning of the operand is also to be found there.

FILE: Local file name

ASYN-MSG:
Specifies whether and which request result leads to an asynchronous termination message. Possible values: ALL, FAIL.

REM: Entry for REMOTE-PARAMETER.
The entry can include more than in this example; the keywords correspond to the recommended abbreviations of the TRANSFER-FILE keywords; the meaning of the operand is also to be found there.

FILE: Remote file name

The following parameters are only output for locally issued requests.

TRANS-ADM:
Transfer admission (here for the remote system. Instead of the triplet user ID, account number and password where appropriate, REMOTE-PROFILE can also be output here if a remote FTAC FT profile is addressed. The equivalent also applies to entries in the local system.

CCSN CCS name used in the local and/or remote system when reading the file.

Example 3

You want to output information about the number of request in each individual processing status. There are three requests in the ACTIVE condition, and five in the WAIT condition. Two requests are still in protocol handling, therefore the sum is 10.

```
/SHFT INF=*SUMMARY

% ACT    WAIT    LOCK    SUSP   HOLD    FIN     TOTAL
%   3       5       0       0      0      0         8
```

4.26 SHOW-FTAC-ENVIRONMENT Display saved admission profiles and sets

*open*FT-AC must be installed to use this command.

The FTAC administrator can use the command SHOW-FTAC-ENVIRONMENT (see page 159) to view admission profiles and sets which have been written in an export file using the command EXPORT-FTAC-ENVIRONMENT. This function is particularly useful before the importing of the admission profiles and sets (see page 161).

This command may not be used by the FTAC user.

```
SHOW-FTAC-ENVIRONMENT

 FROM-FILE = <filename 1..54>
,USER-IDENTIFICATION = *ALL / list-poss(100): <name 1..8>
,SELECT-PARAMETER = *ALL / *PARAMETERS(...)
    *PARAMETERS(...)
        PROFILE-NAME = *ALL / *NONE / list-poss(100): <alphanum-name 1..8>
        ,ADMISSION-SET = *YES / *NO
,INFORMATION = *ONLY-NAMES / *ALL
,OUTPUT = *SYSOUT(...) / *SYSLST(...)
    *SYSOUT(...) / *SYSLST(...)
        LAYOUT = *STD / *CSV
```

Operands

FROM-FILE = <filename 1..54>
Name of the file (not a temporary file) from which the admission profiles and sets are to be displayed. If the file contains invalid data or access to the file is unsuccessful, the command is rejected with the message FTC0103.

USER-IDENTIFICATION =
Here, you enter the user ID whose admission profiles and sets are to be displayed.

USER-IDENTIFICATION = *ALL
The admission profiles and sets of all users are to be displayed.

USER-IDENTIFICATION = list-poss(100): <name 1..8>
The admission profiles and sets of the user IDs specified (maximum 100) are to be displayed.

SELECT-PARAMETER =
This is used to specify whether only admission profiles, only admission sets or both are to be displayed. For the admission profiles, you can specify which ones are to be displayed.

SELECT-PARAMETER = *ALL
All the admission profiles and sets associated with the user ID specified under USER-IDENTIFICATION are to be displayed.

SELECT-PARAMETER = *PARAMETERS(...)
Here, you can specify which of the admission sets associated with the USER-IDENTIFICATION are to be specified.

PROFILE-NAME = *ALL
All admission profiles are displayed.

PROFILE-NAME = *NONE
No admission profiles are displayed.

PROFILE-NAME = list-poss(100): <alphanum-name 1..8>
Only the specified profiles are displayed (maximum 100).

ADMISSION-SET = *YES
All admission sets are displayed.

ADMISSION-SET = *NO
No admission sets are displayed.

INFORMATION =
This specifies the scope of the information to be displayed.

INFORMATION = *ONLY-NAMES
Only the names of the admission profiles are to be displayed.

INFORMATION = *ALL
The entire contents of the admission profiles, excluding any passwords and transfer admissions, are displayed.

OUTPUT =
Defines the output medium.

OUTPUT = *SYSOUT(...)
The output is sent to SYSOUT.

OUTPUT = *SYSLST(...)
The output is sent to SYSLST.

LAYOUT = *STD
The output is formatted using a standard layout that can be easily read by the user.

LAYOUT = *CSV
The output is supplied in CSV (Comma Separated Value) format. This is a widely used tabular format, especially in the PC environment, in which individual fields are separated by a delimiter, which is usually a semicolon ";" (page 104 and 304).

Example

The FTAC administrator Dagobert Duck backs up the admission set and the admission profiles of the user ID DONALD in the file DONALD.FTAC.BKUP.

```
/EXPORT-FTAC-ENVIRONMENT␣TO-FILE=DONALD.FTAC.BKUP,
                         USER-IDENTIFICATION=DONALD
```

A possible short form of this command would be:

```
/EXP-FTAC-ENV␣DONALD.FTAC.BKUP,DONALD
```

As a conscientious FTAC administrator, Dagobert checks if the desired back-up is in the file DONALD.FTAC.SBKUP.

```
/SHOW-FTAC-ENVIRONMENT␣FROM-FILE=DONALD.FTAC.BKUP
```

He receives the following output:

```
                MAX. USER LEVELS                MAX. ADM LEVELS          ATTR
% USER-ID  OBS  OBR  IBS  IBR  IBP  IBF   OBS  OBR  IBS  IBR  IBP  IBF
% DONALD     1    1    0    1    0    0     1    1    0    0    0    0
% OWNER     NAME
% DONALD   *PROFPROD
```

USER-ID and OWNER can be used to determine the user ID with which the admission sets and profiles defined under NAME are associated.

In addition, the maximum security levels set for each user are displayed, as in the command SHOW-FT-ADMISSION-SET. An explanation of these entries can be found in the section for this command (page 244).

Command return codes

(SC2)	SC1	Maincode	Meaning
0	0	FTC0054	No information exists which meets the specified criteria.
0	64	FTC0103	The file is not an FTAC export file or access is not permitted.
0	64	FTC0104	Access to the user ID denied or the user ID doesn't exist.
0	64	FTC0105	Access to the file denied.
0	64	FTC0106	Access to the temporary file denied.
0	64	FTC0156	The command may only be issued by the FTAC administrator.
0	64	FTC0177	The filename entered is unknown.
0	64	FTC0180	The USER-ID entered occurs several times.
0	64	FTC0255	A system error occurred.

SC1/2 = Subcode 1/2 in decimal notation

For additional information, see section "Command return codes" on page 102

OPS variables

The OPS variables of the displayed objects correspond to the variables of the commands SHOW-FT-ADMISSION-SET (see page 247) and SHOW-FT-PROFILE (see page 279).

4.27 SHOW-FT-ADMISSION-SET Display admission sets

Prerequisite for using this command is the use of *open*FT-AC.

You use the SHOW-FT-ADMISSION-SET command to display admission sets. You can output the following information on either SYSOUT or SYSLST:

– if the admission set is privileged (if so, then you are the FTAC administrator).

– if a password is required to use FTAC commands on this user ID. The password itself is not displayed.

– the limiting values for accessible security levels which have been set by the owner of this user ID.

– the limiting values for accessible security levels which have been pre-set by the FTAC administrator.

SHOW-**FT-ADM**ISSION-**S**ET

USER-IDENTIFICATION = ***OWN** / ***ALL** / ***STD** / <alphanum-name 1...8>

,**OUTPUT** = ***SYSOUT**(...) / ***SYSLST**(...)

***SYSOUT**(...) / ***SYSLST**(...)

LAYOUT = ***STD** / ***CSV**

Operands

USER-IDENTIFICATION =
User ID whose admission set you wish to view. FTAC users can only obtain information about their own admission set and the default admission set. The FTAC administrator can obtain information about any admission set.

USER-IDENTIFICATION = *OWN
FTAC outputs your own user ID's admission set.

USER-IDENTIFICATION = *ALL
FTAC outputs the default admission set and the admission set of your own user ID.
For the FTAC administrator, all admission sets are output which differ from the default admission set.

USER-IDENTIFICATION = *STD
FTAC only outputs the default admission set.

USER-IDENTIFICATION = <alphanum-name 1..8>
FTAC outputs the admission set of the user ID indicated. The operand stands for the USER-ID of the specified user ID. The FTAC user can only enter his own user ID here. The FTAC administrator can enter any user ID.

OUTPUT =
Determines the output medium for the information requested.

OUTPUT = *SYSOUT(...)
The output is sent to SYSOUT.

OUTPUT = *SYSLST(...)
The output is sent to SYSLST.

LAYOUT = *STD
The output is formatted using a standard layout that can be easily read by the user.

LAYOUT = *CSV
The output is supplied in CSV (Comma Separated Value) format. This is a widely used tabular format, especially in the PC environment, in which individual fields are separated by a delimiter, which is usually a semicolon ";" (see page 104 and 297).

Example

Dagobert Duck, the FTAC administrator of the Duck Bank, wants to obtain information about the admission sets in his system. He enters the command

```
/SHOW-FT-ADMISSION-SET␣USER-IDENTIFICATION=*ALL
```

Short form:

```
/SHOW-FT-AD␣*ALL
```

and receives the following output:

```
%                  MAX. USER LEVELS      MAX. ADM LEVELS         ATTR
% USER-ID OBS OBR IBS IBR IBP IBF  OBS OBR IBS IBR IBP IBF
% *STD     10  10  10  10   0   0   10  10  10  10   0   0
% DAGO    100 100   0   0   0*  0* 100 100   0   0   0*  0*   PRIV
% DAISY    50  50  10* 50  50  50   50  50  50  50  50  50    PW
% DANIEL    0  10   0   0   0   0   10  10   0   0   0   0    PW
% DONALD   50 100   0  10*  0   0   50 100  10  50   0   0
```

These can be explained as follows:

The user ID of each admission set is in the column USER-ID. In this example, there is a default admission set as well as admission sets for the user IDs DAGO, DAISY, DANIEL and DONALD.

The column ATTR indicates the privileged admission set. We can see that DAGO is the FTAC administrator.

The column ATTR also indicates whether an FTAC password has been defined (with PW). DAGOBERT, DAISY and DANIEL have done this to prevent others from using FTAC commands on their user ID which could be used to make modifications.

In the six columns under MAX-USER-LEVELS, the limiting values are output which the FTAC users have set for their admission sets. The six columns under MAX-ADM-LEVELS show the limiting values which the FTAC administrator has set. The smaller of the two values indicates up to which security level the owner of the admission set may use each basic function. The basic functions are abbreviated in the output as follows:

OBS = **O**UT**B**OUND-**S**END

OBR = **O**UT**B**OUND-**R**ECEIVE

IBS = **I**N**B**OUND-**S**END

IBR = **I**N**B**OUND-**R**ECEIVE

IBP = **I**N**B**OUND-**P**ROCESSING

IBF = **I**N**B**OUND-**F**ILEMANAGEMENT

The default admission set is configured such that it permits file transfers with systems which have the security level of 10 or lower, but does not permit any follow-up processing initiated by external sources (IBP=0). DAGO may contact all available partner systems (OBS=100,OBR=100), but does not permit any file transfer accesses from outside onto his user ID (IBS=0,IBR=0,IBP=0).
The user ID is permitted to communicate with all partner systems with the security level of 50, according to the FTAC administrator’s specifications. To better protect her files from strangers, DAISY has only made the function “inbound send” available to partner systems with the security level f 10 or lower.
The user ID DANIEL is heavily protected. Only files from partner systems with a maximum security level of 10 may be requested. A * after a number indicates that this value was taken from the default admission set and will change if any modifications are made to the default admission set.

Command return codes

(SC2)	SC1	Maincode	Meaning
0	64	FTC0052	The information output was interrupted.
0	64	FTC0152	The user ID entered is not the user's own ID.
0	64	FTC0181	The FT profile name entered occurs several times.
0	64	FTC0255	A system error occurred.

SC1/2 = Subcode 1/2 in decimal notation
For additional information, see section "Command return codes" on page 102

OPS variables

Element	Type	Output
USER-ID	String	
USER-LEV	Struct	
.MAX-OBS	Integer	
.MAX-OBS-STD	String	*YES / *NO
.MAX-OBR	Integer	
.MAX-OBR-STD	String	*YES / *NO
.MAX-IBS	Integer	
.MAX-IBS-STD	String	*YES / *NO
.MAX-IBR	Integer	
.MAX-IBR-STD	String	*YES / *NO
.MAX-IBP	Integer	
.MAX-IBP-STD	String	*YES / *NO
.MAX-IBF	Integer	
.MAX-IBF-STD	String	*YES / *NO
ADM-LEV	Struct	
.MAX-OBS	Integer	
.MAX-OBS-STD	String	*YES / *NO
.MAX-OBR	Integer	
.MAX-OBR-STD	String	*YES / *NO
.MAX-IBS	Integer	
.MAX-IBS-STD	String	*YES / *NO
.MAX-IBR	Integer	
.MAX-IBR-STD	String	*YES / *NO

Element	Type	Output
.MAX-IBP	Integer	
.MAX-IBP-STD	String	*YES / *NO
PRIV	String	*YES / *NO
.MAX-IBF	Integer	
.MAX-IBF-STD	String	*YES / *NO
PASSWORD	String	*YES / *NO

4.28 SHOW-FT-LOGGING-RECORDS (FTSHWLOG) Display logging records

With the SHOW-FT-LOGGING-RECORDS command (o FTSHWLOG), you can obtain information on all FT requests logged by *open*FT.
The logging records are marked as FT or FTAC accordingly, enabling you to identify the type of logging record.

FT logging records

As FT administrator, you can display all the FT logging records in the system. The FT users can only display FT logging records entered under their own ID.
*open*FT only writes a logging record if the request has already reached the FT phase and always when follow-up processing has been started on the inbound side.
If no options are specified, *open*FT outputs the most recent logging record. If options are specified, *open*FT outputs all logging records up to the time specified in the command in reverse chronological order, i.e. starting from the most recent record to the oldest record.

Command execution may take several minutes, depending on the size of the log file. The output can be interrupted using the K2 key.

There are two types of output: short output and long output.

FTAC logging records

With FTAC functionality, SHOW-FT-LOGGING-RECORDS can be used to display the FTAC logging records. FT and FTAC administrators can view all FT and FTAC logging records respectively. If the access check was positive and *open*FT accepted the request, a second logging record is created in *open*FT, indicating whether the request was completed successfully, and if not, why it was terminated.

A precise description of output can be found starting on page 258.

SHOW-FT-LOGGING-**REC**ORDS / **FTSHWLOG**

```
SELECT = *OWN / *ALL / *PARAMETERS(...)

  *PARAMETERS(...)
    LOGGING-ID = *ALL / <integer 1..99999999> / *INTERVAL(...)
      *INTERVAL(...)
        FROM = 1 / <integer 1..99999999>
        ,TO = *HIGHEST-EXISTING / <integer 1..99999999>
    ,OWNER-IDENTIFICATION = *OWN / *ALL / <name 1..8>
    ,CREATION-TIME = *INTERVAL(...) / *DAYS(...)
      *INTERVAL(...)
        FROM = 1970-01-01(...) / <date 8..10>(...)
          <date 8..10>(...)
            TIME = 00:00 / <time 1..8>
        ,TO = *TOMORROW(...) / *TODAY(...) / <date 8..10>(...)
          <date 8..10>(...)
            TIME = 00:00 / <time 1..8>
      *DAYS(...)
        NUMBER = <integer 1..1000>
    ,RECORD-TYPE = *ALL / *PARAMETERS(...)
      *PARAMETERS(...)
        FT = *TRANSFER-FILE / *NONE / list-poss(1): *TRANSFER-FILE
        FTAC = (*TRANSFER-FILE, *READ-FILE-ATTRIBUTES, *DELETE-FILE,
                *CREATE-FILE, *MODIFY-FILE-ATTRIBUTES
                *READ-DIRECTORY, *MOVE-FILE, *CREATE-DIRECTORY,
                *DELETE-DIRECTORY,*MODIFY-DIRECTORY) / *NONE /
                list-poss(10): *TRANSFER-FILE / *READ-FILE-ATTRIBUTES / *DELETE-FILE /
                *CREATE-FILE / *MODIFY-FILE-ATTRIBUTES / *READ-DIRECTORY /
                *MOVE-FILE / *CREATE-DIRECTORY / *DELETE-DIRECTORY /
                *MODIFY-DIRECTORY
    ,INITIATOR = (*LOCAL, *REMOTE) / list-poss(2): *LOCAL / *REMOTE
    ,PARTNER = *ALL / <text 1..200 with-low>
    ,FILE = *ALL / <filename 1..54> / <partial-filename 2..53> / <c-string 1..512 with-low> /
                *DIRECTORY(...) / *POSIX(NAME=<posix-pathname 1..219>)
      *DIRECTORY(...)
        NAME = *ALL / <partial-filename 2..53> / <c-string 1..512 with-low>
    ,REASON-CODE = *ALL / *FAILURE / <text 1..4>
,NUMBER = 1 / *ALL / <integer 1..99999999>
,INFORMATION = *STD / *ALL
,OUTPUT = *SYSOUT(...) / *SYSLST(...)
  *SYSOUT(...) / *SYSLST(...)
    LAYOUT = *STD / *CSV
```

Operands

SELECT =
used to select a group of logging records.

SELECT = *OWN
displays all logging records in the user's own ID.

SELECT = *ALL
displays all users' logging records to the administrator.

SELECT = *PARAMETERS(...)

LOGGING-ID =
Number of the logging record.

LOGGING-ID = *ALL
The number of the logging record is not a selection criterion.

LOGGING-ID = <integer 1..99999999>
Number of the logging record to be output.

LOGGING-ID = *INTERVAL(...)
Range of logging records to be output.

FROM = <integer 1..99999999>
First logging record to be output.

TO = *HIGHEST-EXISTING / <integer 1..99999999>
Last logging record to be output.

OWNER-IDENTIFICATION =
User ID whose logging records are to be displayed.

OWNER-IDENTIFICATION = *OWN
logging records of your user ID are displayed.

OWNER-IDENTIFICATION = *ALL
The logging records of all user IDs are displayed. The FT administrator can thus display the FT logging records of any user ID, and the FTAC administrator can display the FTAC logging records of any user ID.
FT users can also obtain information on logging records relating to their own user ID only.

OWNER-IDENTIFICATION = <name 1..8>
Any user ID whose logging records should be displayed.
FT users may only specify their own ID.

CREATION-TIME =
The range of the logging records to be output, selected by their date or time of creation.

CREATION-TIME = *INTERVAL(...)
The range is specified as a time interval using the date and/or time.

FROM = 1970-01-01(...) / <date 8..10>(...)
Date in the format *yyyy-mm-dd* or *yy-mm-dd*, e.g. 2007-08-18 or 07-08-18 for 18 August, 2007. *open*FT then displays all logging records written after the specified date and time.

TIME = 00:00 / <time 1..8>
Time for the day specified with CREATION-TIME. *open*FT displays all logging records written after the specified time. The time is entered in the format *hh:mm:ss*, e.g. 14:30:10.

TO = *TOMORROW / *TODAY(...) / <date 8..10>(...)
Date in the format *yyyy-mm-dd* or *yy-mm-dd*, e.g. 2007-08-18 or 07-08-18 for 18 August, 2007. *open*FT then displays all logging records written up to the specified date and time.

TIME = 00:00 / <time 1..8>
Time for the day specified with CREATION-TIME. *open*FT displays all logging records written up to the specified time. The time is entered in the format *hh:mm:ss*, e.g. 14:30:10.

CREATION-TIME = *DAYS(NUMBER=<integer 1..1000>)
This field is specified in number of days. All logging sets that were created in the last n calendar days, including today, are output.

RECORD-TYPE =
defines the type of logging record to be displayed.

RECORD-TYPE = *ALL
The record type is not a selection criterion.

RECORD-TYPE = *PARAMETERS(...)
Type of the logging record.

FT = *TRANSFER-FILE / *NONE / list-poss(1): *TRANSFER-FILE
specifies whether or not the FT logging records are to be displayed.

FTAC =
(*TRANSFER-FILE,*READ-FILE-ATTRIBUTES,*DELETE-FILE,*CREATE-FILE, *MODIFY-FILE-ATTRIBUTES,*READ-DIRECTORY,*MOVE-FILE, *CREATE-DIRECTORY,*DELETE-DIRECTORY,*MODIFY-DIRECTORY) / *NONE / list-poss(10): *TRANSFER-FILE / *READ-FILE-ATTRIBUTES / *DELETE-FILE / *CREATE-FILE / *MODIFY-FILE-ATTRIBUTES / *READ-DIRECTORY / *MOVE-FILE / *CREATE-DIRECTORY / *MOVE-DIRECTORY / *DELETE-DIRECTORY
specifies whether or not FTAC logging records are to be displayed. If they are to be displayed, the FT function for which the FTAC logging records are to be displayed can also be specified. The following values are possible:

*TRANSFER-FILE
All logging records for the function “Transfer files” are displayed.

*READ-FILE-ATTRIBUTES
All logging records for the function “Read file attributes” are displayed.

*DELETE-FILE
All logging records for the function “Delete files” are displayed.

*CREATE-FILE
All logging records for the function “Create files” are displayed.

*MODIFY-FILE-ATTRIBUTES
All logging records for the function “Modify file attributes” are displayed.

*READ-DIRECTORY
All logging records for the function “Read file directory” are displayed.

*MOVE-FILE
All logging records for the function “Copy and delete files” are displayed.

*CREATE-DIRECTORY
All logging records for the function “Create directory” are displayed.

*DELETE-DIRECTORY
All logging records for the function “Delete directory” are displayed.

*MODIFY-DIRECTORY
All logging records for the function “Modify directory” are displayed.

INITIATOR =
selects the logging records according to the initiator.

INITIATOR = (*LOCAL,*REMOTE)
The initiator is not a selection criterion.

INITIATOR = *LOCAL
Only those logging records that belong to requests issued locally are displayed.

INITIATOR = *REMOTE
Only those logging records belonging to requests made from a remote system are displayed.

PARTNER =
The partner system.

PARTNER = *ALL
The partner system is not a selection criterion.

PARTNER = <text 1..200 with-low>
Name or address of the partner system for which the logging records are to be displayed. For more information on address specifications, see section “Specifying partner addresses” on page 39.

FILE =
File name.

FILE = *ALL
The file name is not a selection criterion.

FILE = <filename 1..54> / <c-string 1..512 with-low> /
***POSIX(NAME = <posix-pathname 1..219>)**
Fully qualified name of the files for which you wish to view the logging records.

FILE = <partial-filename 2..53>
Partially qualified name of the files for which you want to view the logging records.

FILE =*DIRECTORY(...)
Name of the directory.

***DIRECTORY(...)**
The directory specification relates to the corresponding specification in the SHOW-REMOTE-FILE-ATTRIBUTES command (see “*open*FT for BS2000 User Guide”).

NAME = *ALL
The directory is not a selection criterion.

NAME = <partial-filename 2..53> / <c-string 1..512 with-low>
Name of the directory.

REASON-CODE =
Selection by the reason code of the logging records.

REASON-CODE = *ALL
The reason code is not a selection criterion; all records are output.

REASON-CODE = *FAILURE
All logging records with error codes are output.

REASON-CODE = <text 1..4>
Defines the logging records to be output by the error codes. Leading zeros can be omitted (e.g. 14 for FTR0014).

NUMBER =
Maximum number of logging records.

NUMBER = 1 / <integer 1..99999999>
The maximum number of logging records that are to be displayed. The default value is 1.

NUMBER = *ALL
All logging records are displayed.

INFORMATION =
Scope of the requested information.

INFORMATION = *STD
The logging records are displayed in a standard format (see page 256).

INFORMATION = *ALL
The logging records are displayed in a detailed format (see page 256).

OUTPUT =
Determines the output medium.

OUTPUT = *SYSOUT(...)
The output is sent to SYSOUT.

OUTPUT = *SYSLST(...)
The output is sent to SYSLST.

LAYOUT = *STD
The output is formatted using a standard layout that can be easily read by the user.

LAYOUT = *CSV
The output is supplied in CSV (Comma Separated Value) format. This is a widely used tabular format, especially in the PC environment, in which individual fields are separated by a delimiter, which is usually a semicolon ";" (see page 104 and 298).

4.28.1 Description of the output fields

Short output form of an FT logging record (example)

```
/SHOW-FT-LOGGING-RECORDS NUMBER=2

%TYP LOGG-ID  TIME     RC    PARTNER  INITIATOR INIT USER-ADM FILENAME
%2006-02-26
%T       5333 14:18:24 0014 <G133H301 FT2V292   1TCL FT2V292  TEST2
%T       5284 14:08:12 0000 >G133H301 FT2V292   1TCL FT2V292  TEST1
```

Explanation

Name	Explanation	
TYP (column 1)	Specifies if it is an FT or FTAC log record.T indicates the FT logging record, C indicates the FTAC logging records and P indicates the FTP logging set.	
TYP (columns 2-3)	Definition of FT function:	
	␣	transfer file
	V	transfer file and delete send file (only inbound possible)
	A	read file attributes
	D	delete file
	C	create file
	M	modify file attributes
	R	read directory
	CD	create director
	MD	modify directory
	DD	delete directory
	L	login (inbound FTP access)
LOGG-ID	Number of the logging record (max. 8 numbers)	
TIME	Time when the logging record was written	
RC	Reason Code. Indicates if a request was successfully executed, or if not, why it was rejected or terminated. If an FT request is rejected for “FTAC reasons” (e.g. 0014), the exact reason behind the termination can be found in the FTAC logging record of the system that rejected the request. Further information on the reason code can be obtained using the BS2000 command HELP-MSG-INFORMATION (FTCxxxx for FTAC type or FTRxxxx for FT type)	

Name	Explanation	
PARTNER	Provides information about the partner system. The output includes the symbolic name under which the system administrator has entered the partner system in the partner list. In the short form, the partner system name is given an identifier from which you can determine the request direction.	
	>	The request direction is to the partner system. This direction is specified for a – send request , i.e. the data is transferred to the partner – request to view remote file attributes – request to view remote directories
	<	The request direction is to the local system. This direction is specified for a – receive request, i.e.the data is transferred to the local system – request to modify remote file attributes[1] – request to delete remote files
INITIATOR	Request initiator; if initiative is from remote system: *REMOTE	
INIT	TSN from which the request came. If the INITIATOR was *REMOTE, the field is empty.	
USER-ADM	User ID in the local system used by the requests	
FILENAME	Filename in local system	

[1] When modifying the access rights of a file from an FTAM partner system, two logging records are written. In this case, no direction is specified before the PARTNER output.

Long output form (example)

```
/SHOW-FT-LOGGING-RECORDS,LOGGING-ID=54,INF=*ALL
%LOGGING-ID  = 00003705  RC      = 2072        TIME     = 2007-07-19 13:43:11
%   TRANS    = TO        REC-TYPE= FT          FUNCTION = TRANSFER-FILE
%   PROFILE  =           PCMD    = NONE        STARTTIME=
%   TRANS-ID = 69158     WRITE   = REPLACE     REQUESTED= 2006-07-19 13:41:32
%   TRANSFER =         0 kB                    CCS-NAME =
%   INITIATOR= TSOS                            INITSN   = 12ZI
%   USER-ADM = TSOS
%   PARTNER  = HUGO
%   FILENAME = BULLETIN
%LOGGING-ID  = 00003704  RC      = 0000        TIME     = 2007-07-19 13:41:32
%   TRANS    = TO        REC-TYPE= FTAC        FUNCTION = TRANSFER-FILE
%   PROFILE  =           PRIV    =
%   INITIATOR= TSOS                            INITSN   = 12ZI
%   USER-ADM = TSOS
%   PARTNER  = HUGO
%   FILENAME = BULLETIN
```

Explanation of long output form (column-wise)

Name	**Explanation**	
LOGGING-ID	Number of the logging record (max. 8 numbers	
TRANS	Transfer direction:	
	TO	The request direction is to the partner system. This direction is specified for a – send request, i.e. the data are transferred to the partner – request to view remote file attributes – request to view remote directories
	FROM	The request direction is to the local system (inbound). This direction is specified for a – receive request, i.e. the data are transferred to the local system – request to modify remote file attributes[1] – request to delete remote files
	BOTH	File management request with two-way data transfer.
PROFILE	Name of the profile to be used for the transfer (empty in the FT logging record)	
TRANS-ID	Transfer ID number	
TRANSFER	Amount of data transferred	

Name	Explanation	
SEC-OPTS	Security options and encryption algorithms used. This line is only output if at least one of the options is used.	
	ENCR	Encryption of the request queue
	DICHK	Data integrity check of the request queue
	DENCR	Encryption of data content during the transfer
	DDICHK	Data integrity check of the file data to be transferred
	LAUTH	Authentication of the local system on a partner
	RAUTH	Authentication of the partner on a local system
	RSA-nnnn	Length of the RSA key
	DES / AES	Encryption algorithm used
INITIATOR	Request initiator; if initiative is from remote system: *REMOTE	
PARTNER	Provides information about the partner system. The output includes the symbolic name under which the system administrator has entered the partner system in the partner list.	
FILENAME	Filename in local system	
RC	Reason-Code. Indicates if a request was successfully executed, or if not, why it was rejected or terminated. If an FT request is rejected for “FTAC reasons” (e.g. 0014), the exact reason behind the termination can be found in the FTAC logging record of the system that rejected the request. Further information on the reason code can be obtained using the BS2000 command HELP-MSG-INFORMATION (FTCxxxx for FTAC type or FTRxxxx for FT type).	
REC-TYPE	Specifies if this is an FT, FTAC or FTAC(FTP) logging record.	
PCMD	Status of follow-up processing:	
	NONE	No follow-up processing defined.
	STARTED	Follow-up processing was started.
	NOT-STARTED	Follow-up could not be started.
PRIV	specifies if the admission profile is privileged.	
WRITE	Write rules	
	NEW	A new file is created. If a file with the same name already exists, the transfer will be cancelled.
	EXT	An existing file is extended and stored as new.
	REPLACE	An existing file is extended.
INITTSN	TSN from which the request came. If the INITIATOR was REMOTE the field is empty.	
USER-ADM	User ID in the local system used by the requests	

Name	Explanation
TIME	Time when the logging record was written
FUNCTION	Definition of FT function:
	– TRANSFER-FILE: transfer file – MOVE-FILE: transfer file and delete send file (only inbound possible) – READ-FILE-ATTRIBUTES: read file attributes – DELETE-FILE: delete file – CREATE-FILE: create new file – MODIFY-FILE-ATTRIBUTES: modify file attributes – READ-DIRECTORY: read directory – CREATE-DIRECTORY: create directory – MODIFY-DIRECTORY: modify directory – DELETE-DIRECTORY: delete directory – LOGIN: inbound FTP access
STARTTIME	Time request was started
STORETIME	Time request was accepted (inbound)
REQUESTED	Time request was accepted (outbound)
CCS-NAME	Name of the character set, used for code conversion as necessary.

[1] When modifying the access rights of a file from an FTAM partner system, two logging records are written. In this case, no direction is specified before the PARTNER output.

Command return codes

(SC2)	SC1	Maincode	Meaning
0	0	CMD0001	No log records available for the selection criteria.
33	32	CMD0221	Request rejected. Internal error.
36	32	CMD0221	Request rejected. Request data inconsistent.
83	32	CMD0221	Internal error.
88	64	CMD0221	Error during OPS generation.
36	64	FTR1036	User not authorized for other user IDs.
2	0	FTR2225	Information output cancelled.

SC1/2 = Subcode 1/2 in decimal notation

For additional information, see section “Command return codes” on page 102

OPS variables

The following table shows the OPS variables for the command SHOW-FT-LOGGING-RECORDS with the operand INF=*ALL, the underlined values are valid for the output with the operand INF=*STD.

Element	Type	Output
LOG-ID	Integer	
REASON-CODE [1]	Integer	
LOG	Struct	
.DATE	String	yyyy-mm-dd
.TIME	String	hh:mm:ss
INIT-USER-ID	String	USER-ID of request initiator / *REM
INIT-TSN [2]	String	TSN of request initiator
PARTNER-NAME	String	
TRANS-DIRECT	String	*TO-PARTNER / *FROM-PARTNER / *NOT-SPECIFIED
REC-TYPE	String	*FT / *FTAC
FUNC	String	*TRANS-FILE / *READ-FILE-ATTR / *DEL-FILE / *CRE-FILE / *MOD-FILE-ATTR / *READ-DIR / *CRE-DIR / *MOD-DIR / *DEL-DIR / *MOVE-FILE / *LOGIN
USER-ADMIS	String	
WRITE-MODE	String	*REPL-FILE / *NEW-FILE / *EXT-FILE
RESULT-PROCESS	String	*NONE / *STARTED / *NOT-STARTED
START	Struct	
.DATE	String	yyyy-mm-dd
.TIME	String	hh:mm:ss
TRANS-ID	Integer	
STORE	Struct	
.DATE	String	yyyy-mm-dd
.TIME	String	hh:mm:ss
BYTE-NUM	String	
PRIVIL [3]	String	*NO / *YES
PROF-NAME [3]	String	
F-NAME	String	

Element	Type	Output
SEC	Struct	
.PROT.ENC	String	*NO / *YES
.PROT.INT-CHECK	String	*NO / *YES
.USER-DATA.ENC	String	*NO / *YES
.USER-DATA.INT-CHECK	String	*NO / *YES
.LOC-AUTH	String	*NO / *YES
.REM-AUTH	String	*NO / *YES
RSA-KEY-LEN	Integer	
SYMM-ENC-ALG	String	*DES / *AES

[1] The reason code is always given in decimal form. To determine the meaning of FTAC logging records using the manual, the value must be converted to hexadecimal form.

[2] For INIT-USER-ID=*REM, INIT-TSN is not assigned.

[3] Only for REC-TYPE=*FTAC and specification of a profile.

Example

The FT administrator wants to display all logging records that were created for the user ID *Meier* and logged between 01.01.2007 and 30.06.2007.

```
/SHOW-FT-LOGGING-RECORDS SELECT=*PARAMETERS(OWNER-IDENTIFICATION=Meier, -
/                        CREATION-TIME=*INTERVAL(FROM=2007-01-01(00:00), -
/                           TO=2007-06-30(23:59))),NUMBER=*ALL
```

You want to see the first record of the output in detail.

```
/SHOW-FT-LOG-REC (OWN=Meier,CRE-TIME=*INTERVAL(FROM=2007-01-01(00:00),  -
/                           TO=2007-06-30(00:00))),INF=*ALL
```

4.29 SHOW-FT-OPTIONS (FTSHWOPT) Display operating parameters

The command SHOW-FT-OPTIONS (also FTSHWOPT) can be used at any time to obtain the information listed below on the operating parameters of your FT system:

- Maximum number of file transfer requests in the request queue,
- Maximum number of tasks that can process file transfer requests concurrently,
- Maximum number of transport connections that can be reserved simultaneously for file transfer requests,
- Maximum size of a transport unit in bytes,
- Default setting for the FTAC security level of the partner systems,
- Status of extended authentication check (on or off),
- Which type of trace is switched on,
- Whether the logging is switched on,
- Maximum lifetime of FT requests in the request file,
- The name of the BCAM host. The default value is *NONE, i.e. the real BCAM host is used.
- The instance ID. After installation, this field also contains the name of the BCAM host.
- Whether and which SNMP traps are activated or deactivated,
- Whether and which console traps are activated or deactivated.

SHOW-FT-OPTIONS / **FTSHWOPT**
OUTPUT = ***SYSOUT**(...) / ***SYSLST**(...) ***SYSOUT**(...) / ***SYSLST**(...) **LAYOUT** = ***STD** / ***CSV** / ***BS2-PROC** / ***ZOS-PROC**

Operands

OUTPUT =
Establishes the output medium.

OUTPUT = *SYSOUT(...)
Output takes place on SYSOUT.

OUTPUT = *SYSLST(...)
Output takes place on SYSLST.

LAYOUT = *STD
The output is put into a user-friendly form for reading.

LAYOUT = *CSV
The output takes place in Comma Separated Value format. This is a special tabular format, widely used in the PC world, where the individual fields are separated by semicolons ";" (see page 104 and 300).

LAYOUT = *BS2-PROC
The operating parameters are output as a command sequence. This can be called as an SDF procedure at BS2000/OSD systems in order to recreate the identical operating parameters.

LAYOUT = *ZOS-PROC
The operating parameters are output as a command sequence. This can be called as a Clist procedure at z/OS systems in order to recreate the identical operating parameters.

The current values of the operating parameters are listed as follows:

```
STARTED PROC-LIM CONN-LIM RQ-LIM MAX-RQ-LIFE TU-SIZE KEY-LEN
  *YES      4       16      100   30          65535    768
PTN-CHK DYN-PART SEC-LEV  FTAC-LOG FT-LOG
  STD      *ON       1       ALL     ALL
OPENFT-APPL      FTAM-APPL       FTP-PORT
*STD             *STD            21
HOST-NAME        IDENTIFICATION
*NONE            HOSTE.FUSI.NET

TRAP: SS-STATE FT-STATE PART-STATE PART-UNREA RQ-STATE TRANS-SUCC TRANS-FAIL
CONS     ON       ON        ON         ON        ON        ON         ON
SNMP    OFF      OFF       OFF        OFF       OFF       OFF        OFF

TRACE: SWITCH PARTNER-SELECTION    REQUEST-SELECTION    OPTIONS
        OFF   ALL                  ALL                  NONE
```

Command return codes

(SC2)	SC1	Maincode	Meaning
83	32	CMD0221	Internal error
88	64	CMD0221	Error during ops generation
35	64	FTR1035	User not authorized for this command.
2	0	FTR2225	Information output canceled

SC1/2 = Subcode 1/2 in decimal notation

For additional information, see section “Command return codes” on page 102

OPS variables

The following table shows the OPS variables for the command SHOW-FT-OPTIONS.

Element	Type	Output
REQ-LIM	Integer	
TASK-LIM	String	
CONN-LIM	Integer	
TRANSPORT-UNIT-SIZE	Integer	
PARTNER-CHECK	String	*STD / *TRANSP-ADDR
SEC-LEV	String	Value / *BY-PARTNER-ATTRIBUTES
TRACE	Struct	
.STATE	String	*ON / *OFF
.OUT	String	*FILE / leer
.PARTNER-SEL	Struct	
.OPENFT	String	*YES / *NO
.FTAM	String	*YES / *NO
.FTP	String	*YES / *NO
.REQUEST-SEL	Struct	
.ASYNC	String	*YES / *NO
.SYNC	String	*YES / *NO
.LOCAL	String	*YES / *NO
.REMOTE	String	*YES / *NO
.OPTIONS	Struct	
.BULK-DATA	String	*YES / *NO
LOG	Struct	
.TRANS-F	String	*ON / *OFF / *FAILURE
.FTAC	String	*ON / *REJECTED / *MODIFICATIONS
MAX-REQ-LIFETIME	String	*UNLIMITED / max-request-lifetime
SNMP-TRAPS	Struct	
.SUBSYSTEM-STATE	String	*OFF / *ON
.FT-STATE	String	*OFF / *ON
.PARTNER-STATE	String	*OFF / *ON
.PARTNER-UNREACHABLE	String	*OFF / *ON
.REQUEST-QUEUE-STATE	String	*OFF / *ON
.TRANSFER-SUCCESS	String	*OFF / *ON

Element	Type	Output
.TRANSFER-FAILURE	String	*OFF / *ON
CONSOLE-TRAPS[1]	String	*OFF / *ON
CONS-TRAPS	Struct	
.SUBSYSTEM-STATE	String	*OFF / *ON
.FT-STATE	String	*OFF / *ON
.PARTNER-STATE	String	*OFF / *ON
.PARTNER-UNREACHABLE	String	*OFF / *ON
.REQUEST-QUEUE-STATE	String	*OFF / *ON
.TRANSFER-SUCCESS	String	*OFF / *ON
.TRANSFER-FAILURE	String	*OFF / *ON
HOST-NAME	String	Name of the BCAM host
IDENTIFICATION	String	ID of the local *open*FT instance
DYNAMIC-PARTNERS	String	*ON / *OFF
KEY-LENGTH	Integer	Value
STARTED	String	*YES / *NO
OPENFT-APPLICATION	String	*STD / value
FTAM-APPLICATION	String	*STD / value
FTP-PORT	String	*NONE / value
OPENFT-STD	String	*STD / value

[1] Now only support for reasons of compatibility. The value is only set if all the console traps are activated (*ON) or if all the console traps are deactivated (*OFF).

Meaning of the output fields

REQ-LIM
is the maximum number of FT requests that can be entered at the same time in the request queue of the local system. The value can be modified by means of the REQUEST-LIMIT operand in the MODIFY-FT-OPTIONS command.
Default value: 2000.

PROC-LIM
specifies the maximum number of tasks that can be reserved simultaneously for the execution of FT requests.
Default value: 2.

CONN-LIM
specifies the maximum number of transport connections that can be reserved for the execution of FT requests. Since each transport connection can only process one request at a time, CONNECTION-LIMIT also defines the maximum number of requests that can be processed simultaneously. One third of the transport connections are reserved for requests from remote systems. The value of CONNECTION-LIMIT is defined by the CONNECTION-LIMIT operand in the MODIFY-FT-OPTIONS command.
Default value: 16

TRANSPORT-UNIT-SIZE
specifies the maximum size of a transport unit in bytes. The value is defined with the TRANSPORT-UNIT-SIZE operand in the MODIFY-FT-OPTIONS command. The load placed on the transport system by *open*FT can be controlled using this operand.
Default value: 65565

PARTNER-CHECK
defines whether or not the authentication check is activated.
Default value: STD.

SECURITY-LEVEL
specifies the local default value for the security level of the partner systems. This operand is only effective if FTAC functionality is being used. An important part of the access protection functions provided by this product lies in the allocation of security levels to remote systems. To this end, each system is allocated a security level designated using an integer in the range 1 to 100. A default value is set by using the SECURITY-LEVEL operand in the MODIFY-FT-OPTIONS command. This value can then be defined for an individual system by specifying SECURITY-LEVEL=STD in the ADD-FT-PARTNER command.
Default value: for partners in the partner list: *STD (the security level depends on the type of partner).
In the case of dynamic partners, the value *BY-PARTNER-ATTRIBUTES is always output.

TRACE
Specifies whether and which traces are activated and what trace scope is set. Possible values for the trace status are *ON/*OFF.
The scope of the traces is specified on the basis of the monitored partners and request types. Possible values for the trace scope are *YES/*NO.
You activate and deactivate with /MODIFY-FT-OPTIONS TRACE=....
Default value: OFF

LOG
Specifies the status of FT logging (TRANS-F) or FTAC logging (FTAC).
Possible values for FT logging: *ON / *OFF / *FAILURE
Possible values for FTAC logging: *ON / *REJECTED / *MODIFICATIONS
Change with MODIFY-FT-OPTIONS LOGGING=....
Default value: ON

MAX-REQUEST-LIFETIME
Specifies the maximum number of days that an FT request is stored in the request file after its start time. When this period expires, the FT request is automatically removed from the request file. The value is defined in the MAX-REQUEST-LIFETIME operand of the MODIFY-FT-OPTIONS command.
Default value: 30

SNMP-TRAPS
Specifies whether the individual SNMP traps are activated or deactivated. SNMP traps are generated in order to display specific events which FT sub-agents route to an SNMP Management Station if available. The traps are defined using the SNMP-TRAPS operand in the MODIFY-FT-OPTIONS command.
The possible values for each trap are ON or OFF.
Default value: OFF

CONS-TRAPS
For each individual event, specifies whether console traps are output. The traps are defined using the CONSOLE-TRAPS operand in the MODIFY-FT-OPTIONS command. The possible values are ON or OFF.
Default value: ON

HOST-NAME
The name of the BCAM host that was set using MODIFY-FT-OPTIONS or real BCAM host.

IDENTIFICATION
ID of the local *open*FT instance. *open*FT partners as of V8.1 use this instance ID to administer the resources for their *open*FT instance.

DYNAMIC-PARTNERS
Specifies whether dynamic partners are permitted (*ON) or not (*OFF).
Default value: ON

KEY-LENGTH
Specifies the current length of the RSA key.

STARTED
Specifies whether *open*FT is active

OPENFT-APPLICATION
Specifies the port number used by the local *open*FT server. *STD means that the default port number 1100 is used.

FTAM-APPLICATION
Specifies the port number used by the local FTAM server. *STD means that the default port number 4800 is used

FTP-PORT
Specifies the port number used by the local FTP server. NONE means that the FTP server is deactivated.

OPENFT-STD
Specifies the port number used to address *open*FT partners if these are addressed via their host names without any port number specification.
*STD means that the default port number 1100 is used.

Example

Default of the SHOW-FT-OPTIONS command in BS2000, i.e. the operating parameters have not been modified since installation.

```
/sh-ft-opt
STARTED PROC-LIM CONN-LIM RQ-LIM MAX-RQ-LIFE TU-SIZE KEY-LEN
  *YES      2       16     2000      30       65535    768
PTN-CHK DYN-PART SEC-LEV  FTAC-LOG FT-LOG
  STD      *ON       1       ALL     ALL
OPENFT-APPL      FTAM-APPL       FTP-PORT
*STD             *STD            NONE
HOST-NAME        IDENTIFICATION
*NONE            PROTEUS3

TRAP: SS-STATE FT-STATE PART-STATE PART-UNREA RQ-STATE TRANS-SUCC TRANS-
FAIL

CONS    OFF      OFF       OFF        OFF       OFF       OFF        OFF
SNMP    OFF      OFF       OFF        OFF       OFF       OFF        OFF

TRACE: SWITCH PARTNER-SELECTION    REQUEST-SELECTION    OPTIONS
        OFF   ALL                   ALL                  NONE
```

4.30 SHOW-FT-PARTNERS (FTSHWPTN) Display partner systems

The SHOW-FT-PARTNERS (or FTSHWPTN) command is used to obtain the following information on partner systems included in the partner list:

- the names of the remote systems in the partner list,
- the status of the requests with the remote systems (activated or deactivated),
- the security level assigned to the remote system (this security level applies only if FTAC functionality is used. The information can then also be obtained using the SHOW-FT-RANGE command),
- the number of not yet completed file transfer requests submitted in the local system,
- the number of file transfer requests submitted in the remote systems for the local system,
- the partner address.

SHOW-FT-PARTNERS / **FTSHWPTN**

PARTNER = ***ALL** / <text 1..200 with-low>

,**OUTPUT** = ***SYSOUT**(...) / ***SYSLST**(...)

***SYSOUT**(...) / ***SYSLST**(...)
LAYOUT = ***STD** / ***CSV** / ***BS2-PROC** / ***ZOS-PROC**

,**STATE** = ***ALL** / ***ACTIVE** / ***DEACT** / ***INSTALLATION-ERROR** / ***NO-CONNECTION** / ***NOT-ACTIVE** / ***AUTOMATIC-DEACTIVATION** / ***INACTIVE-BY-AUTOMATIC-DEACT**

,**INFORMATION** = ***STD** / ***ALL**

Operands

PARTNER =
Specifies the partner system or systems about which information is to be output.

PARTNER = *ALL
Information on all partner systems is output.

PARTNER = <text 1..200 with-low>
The name or address of the partner system or group of partner systems about which information is to be output.
If you enter a name then you have two options:
You can either enter a unique partner name (1 - 8 alphanumeric characters) or a group of partners identified by a 1 to 7-character specification followed by an asterisk (*).
For information on specifying addresses, see section “Specifying partner addresses” on page 39.

OUTPUT =
Determines the output medium.

OUTPUT = *SYSOUT(...)
The output is sent to SYSOUT.

OUTPUT = *SYSLST(...)
The output is sent to SYSLST.

LAYOUT = *STD
The output is formatted using a standard layout that can be easily read by the user.

LAYOUT = *CSV
The output is supplied in CSV (Comma Separated Value) format. This is a widely used tabular format, especially in the PC environment, in which individual fields are separated by a delimiter, which is usually a semicolon “;“ (see page 104 and 302).

LAYOUT = *BS2-PROC
The output is supplied in the form of MODIFY-FT-PARTNER commands, which precisely define the partners involved. This enables the partner entries to be saved for a later reconstruction.
If this output is redirected to a file by using the SYSFILE command, it should be noted that the BS2000 Sysfile Management inserts a blank (i.e., a linefeed character) before each line. The first column of the file must hence be stripped before the procedure generated by this method can be called. We therefore recommend that you use the START-OPENFTPART command, which performs this task for the user.

LAYOUT = *ZOS-PROC
The output takes the form of a command sequence. This can be called as a Clist procedure at z/OS systems in order to recreate the identical information.

STATE =
The scope of the output can be limited by the optional selection criteria in STATE. For an explanation of the selection criteria see page 276.

STATE = *ALL
The output is not limited by selection criteria.

STATE = *ACTIVE
All partner systems in the ACTIVE state are displayed.

STATE = *DEACT
All partner systems in the INACTIVE state are displayed.

STATE = *INSTALLATION-ERROR
All partner systems in the LUNK, RUNK, LAUTH, RAUTH, NOKEY and IDREJ state are displayed.

STATE = *NO-CONNECTION
All partner systems in the NOCON and DIERR state are displayed.

STATE = *NOT-ACTIVE
All partner systems not in the ACTIVE state are displayed.

STATE = *AUTOMATIC-DEACTIVATION
All partner systems are output which were assigned AUTOMATIC-DEACTIVATION.

STATE = *INACTIVE-BY-AUTOMATIC-DEACT
All partner systems are output which were actually deactivated using the option AUTOMATIC-DEACTIVATION.

INFORMATION = *STD / *ALL
Use this operand to control the scope of the information output. On *ALL, expanded address information is output, in addition to the standard information.

Command return codes

(SC2)	SC1	Maincode	Meaning
0	0	CMD0001	No partner available for the selection criteria.
83	32	CMD0221	Internal error.
88	64	CMD0221	Error during OPS generation.
35	64	FTR1035	The user is not authorized to use this command.
45	64	FTR1045	No partner found in partner list.
2	0	FTR2225	Information output cancelled.

SC1/2 = Subcode 1/2 in decimal notation

For additional information, see section “Command return codes” on page 102

OPS variables

The following table shows the OPS variables for the command SHOW-FT-PARTNERS.

Element	Type	Output
PARTNER-NAME	String	
STA	String	*ACTIVE / *INACTIVE / *NO-CONN / *LOC-UNKNOWN /*REM-UNKNOWN / *ACTIVE (AUTO-DEACTIVATE) / *INACTIVE (BY- AUTOMATIC-DEACTIVATION) / *LOC-AUTH-FAIL / *REM-AUTH-FAIL / *DATA-INTEGRITY-ERROR / *NO-KEY / *ID-REJ
SEC-LEV	String	VALUE / *BY-PARTNER-ATTRIBUTES
TRACE [1]	String	*ON / *OFF / *BY-FT-OPT
LOC	Integer	
REM	Integer	
PARTNER-ADDR	String	Value
ADDR-TYPE [2]	String	*OPENFT / *PRESENTATION / *TCP-IP
OPENFT-ADDR [3]	Struct	
.PROCESSOR	String	Value
.ENTITY	String	Value
.NETWORK-ADDR	String	Value
.TRANS-SEL	String	Wet
.PORT	String	port number
.PARTNER-CHECK	String	*FTOPT / *STD / *TRANSP-ADDR / *AUTH
.AUTH-MAND	String	*YES / *NO

Element	Type	Output
.IDENTIFICATION	String	
.SESSION-ROUTING	String	*ID or empty
PRESENTATION-ADDR [4]	Struct	
.NETWORK-ADDR	String	Value
.TRANSPORT-SEL	String	Value
.SESSION-SEL	String	Value
.PRESENTATION-SEL	String	Value
.PORT	String	Value
TCP-IP-ADDR [5]	Struct	
.PORT	String	Value

[1] TRACE is only displayed for *open*FT partners.

[2] Only the address structure corresponding to the ADDR-TYPE element is displayed.

[3] Only applies to *open*FT partners.

[4] Only applies to FTAM partners.

[5] Only applies to FTP partners.

Example

Request information on all remote systems entered in the partner list:

```
/SHOW-FT-PARTNERS INF=*STD
NAME     STATE SECLEV TRACE   LOC    REM CHECK ADDRESS
         ACT   STD    FTOPT     0      0 FTOPT PCNEW
HOSTABS2 ACT B-P-ATTR FTOPT     0      0 FTOPT HOSTABS2
HOSTBBS2 ACT   STD    FTOPT     0      0 FTOPT HOSTBBS2:102
FOREIGN  ACT    10    FTOPT     0      0       ftam://PC3:102.ftam.ftam.ftam
FTAMPC   ACT    30    FTOPT     0      0       ftam://PC2:.sni-ftam
FTAMUX   ACT    30    FTOPT     0      0       ftam://UNIX3
FTIF     ACT    50    FTOPT     0      0       PC4:.fjmftif0.ftmftifpc
PCUSER   ACT    40    FTOPT     0      0 FTOPT %IP123.23.99.120
PC1      ACT    40    FTOPT     0      0 FTOPT PC1
UNIX1    ACT    50    FTOPT     0      0 FTOPT UNIX1
UNIX2    ACT    50    FTOPT     0      0 FTOPT UNIX2:102

/SHOW-FT-PARTNERS INF=*ALL
NAME     STATE SECLEV    TRACE   LOC    REM P-CHK ADDRESS
                                                 ROUTING  IDENTIFICATION
HUGO     ACT   STD       FTOPT     0      0 FTOPT HUGO
                                                         %.HUGO.$FJAM
HOSTABS2 ACT B-P-ATTR    FTOPT     0      0 FTOPT HOSTABS2
                                                         HOSTABS2.FUSI.NET
```

```
HOSTBBS2 ACT   STD       FTOPT     0      0 FTOPT HOSTBBS2:1O2
                                                  HOSTBBS2.FUSI.NET
FOREIGN  ACT    10       FTOPT     0      0
ftam://PC3:1O2.ftam.ftam.ftam
FTAMPC   ACT    30       FTOPT     0      0       ftam://PC2:.sni-ftam
FTAMUX   ACT    30       FTOPT     0      0       ftam://UNIX3
FTIF     ACT    50       FTOPT     0      0       PC4:.fjmftif0.ftmftifpc
PCUSER   ACT    40       FTOPT     0      0 FTOPT %IP123.23.99.120
                                                  %IP123.23.99.120
PC1      ACT    40       FTOPT     0      0 FTOPT PC1
                                                  PC1.FUSI.NET
UNIX1    ACT    50       FTOPT     0      0 FTOPT UNIX1
                                                  UNIX1.FUSI.NET
UNIX2    ACT    50       FTOPT     0      0 FTOPT UNIX2:1O2
                                                  %.UNIX2.$FJAM
```

The information displayed is explained below:

NAME
Specifies the symbolic names of the remote systems entered in the partner list.

STATE
Specifies the status of the partner system.

ACT
The partner system is active.

DEACT
The partner system is deactivated.

NOCON
The transport connection setup failed.

LUNK
The local system is unknown on the remote FT system.

RUNK
The partner system is unknown on the local transport system.

ADEAC
The partner system is active. It is deactivated if the connection cannot be established. This state is only displayed if STATE=*AUTOMATIC-DEACTIVATION has been specified; otherwise, these partner systems are maintained under the ACT status.

AINAC
The partner system was deactivated following several unsuccessful attempts to establish a connection. This status is only possible if STATE=*AUTOMATIC-DEACTIVATION has been specified.

LAUTH
The local system could not be authenticated in the partner system. A current, public key of the local *open*FT instance must be made available to the partner system.

RAUTH
The partner system could not be authenticated in the local system. A current, public key of the partner system must be imported to the SYSKEY library.

DIERR
A data integrity error was detected on the connection to the partner system. This can be due either to an error in the transport system, or to attempts at manipulation along the transfer route. The connection was terminated but the affected request was not (if it is restartable).

NOKEY
The partner does not accept a connection without encryption, but no key is present in the local system. A new key must be created using CREATE-FT-KEY-SET.

IDREJ
The partner or a go-between instance does not accept the instance ID sent from the local system. You must check to see if the local instance ID is consistent with the entry in the partner's partner list.

SECLEV
Defines the security level assigned to the remote system when it was entered in the partner list. These security levels apply only if the FTAC-BS2000 is also implemented. STD stands for the default security level set with the MODIFY-FT-OPTIONS command. It applies to all partner systems which are entered in the partner list and for which nothing has been specifically defined.

TRACE
Describes the trace setting. You may specify the values STD, MIN, OFF and FTOPT (if MODIFY-FT-PARTNER is specified, TRACE=*BY-FT-OPTIONS).

LOC
Gives the number of FT requests that have been submitted in the local system and that address the FT system specified with REMSYS.

REM
Specifies the number of FT requests that have been submitted in the remote FT system and addressed to the local FT system. The remote system is specified in REMSYS.

IDENTIFICATION
Specifies the instance ID of the partner (also see the ADD-FT-PARTNER command, page 105).

ROUTING

Specifies the SESSION-ROUTING-INFO of the partner, where required (also see the ADD-FT-PARTNER command, page 105).

CHECK
Specifies the type of sender checking for the current partner:

FTOPT
The global setting is valid.

T-A
The expanded sender checking is enabled for specific partners.

STD
The expanded sender checking is disabled for specific partners.

AUTH
With the aid of its public key in the SYSKEY library, the partner is subjected to an identity check (“authenticated”) by cryptographic means.

NOKEY
Authentication is required and no key is present for the partner.

AUTHM
Authentication must be used.

ADDRESS
Specifies the partner address under which the remote system can be accessed. For more information on partner addresses, see section “Specifying partner addresses” on page 39.

4.31 SHOW-FT-PROFILE Display admission profile

Prerequisite for using this command is the use of *open*FT-AC.

With the command SHOW-FT-PROFILE, you can obtain information about all the admission profiles in your system. Either the contents of the selected admission profile or only its name can be output. It is not possible to use SHOW-FT-PROFILE to access passwords or transfer admissions defined in the profile! If a transfer admission is forgotten, a new one must be specified using MODIFY-FT-PROFILE.

```
SHOW-FT-PROFILE

 NAME = *ALL / <alphanum-name 1..8>
,SELECT-PARAMETER = *OWN / *PARAMETERS(...)
    *PARAMETERS(...)
        TRANSFER-ADMISSION = *ALL / *NOT-SPECIFIED / <alphanum-name 8..32> /
                                    <c-string 8..32 with-low> / <x-string 15..64> / *SECRET
        ,OWNER-IDENTIFICATION = *OWN / *ALL / <name 1..8>
,INFORMATION = ONLY-NAMES / *ALL
 OUTPUT = *SYSOUT(...) / *SYSLST(...)
    *SYSOUT(...) / *SYSLST(...)
        LAYOUT = *STD / *CSV
```

Operands

NAME =
With NAME, you enter the name of the admission profile you wish to view. NAME accesses the user-wide unique admission profile name.

NAME = *ALL
You wish to view all admission profiles.

NAME = <alphanum-name 1..8>
You wish to view the admission profile with the specified name.

SELECT-PARAMETER =
With SELECT-PARAMETER, you can specify selection criteria for the admission profiles you wish to view.

SELECT-PARAMETER = *OWN
With *OWN, you can view all the admission profiles of which you are the owner. This means that you can view all the admission profiles which are assigned to your user ID.

SELECT-PARAMETER = *PARAMETERS(...)
This structure contains the selection criteria with which you can access your admission profiles.

TRANSFER-ADMISSION =
With TRANSFER-ADMISSION, you can enter the transfer admission defined in an admission profile as a selection criterion. Only the FTAC administrator can enter the user IDs of other users.

TRANSFER-ADMISSION = *ALL
TRANSFER-ADMISSION is not used as a selection criterion.

TRANSFER-ADMISSION = *NOT-SPECIFIED
Only admission profiles for which no transfer admission has been specified are displayed.

TRANSFER-ADMISSION = <alphanum-name 8..32> /
<c-string 8..32 with-low> / <x-string 15..64>
You wish to view the admission profile which can be addressed with this transfer admission.

TRANSFER-ADMISSION = *SECRET
The system prompts you to enter the transfer admission. However, this does not appear on the screen.

OWNER-IDENTIFICATION =
With OWNER-IDENTIFICATION, the FTAC administrator can specify whose admission profiles he wishes to view.
The FTAC user can only access his own admission profiles with this parameter.

OWNER-IDENTIFICATION = *OWN
You wish to view only your own admission profiles.

OWNER-IDENTIFICATION = *ALL
The FTAC administrator can view all admission profiles, regardless of who the owner is.

OWNER-IDENTIFICATION = <name 1..8>
The FTAC administrator can view the admission profiles of any FTAC user with this parameter.

INFORMATION =
With INFORMATION, you determine the scope of information desired.

INFORMATION = *ONLY-NAMES
FTAC only outputs the name of the admission profile and indicates whether it is privileged or blocked. An “∗“ is output for privileged profiles and a “!“ for blocked profiles.

INFORMATION = *ALL
FTAC outputs the contents of the admission profile, excluding any passwords and the transfer admission.

OUTPUT =
With OUTPUT, you can determine the output medium for the information.

OUTPUT = *SYSOUT(...)
The output is sent to SYSOUT.

OUTPUT = *SYSLST(...)
The output is sent to SYSLST.

LAYOUT = *STD
The output is formatted using a standard layout that can be easily read by the user.

LAYOUT = *CSV
The output is supplied in CSV (Comma Separated Value) format. This is a widely used tabular format, especially in the PC environment, in which individual fields are separated by a delimiter, which is usually a semicolon ";" (see page 104 and 303).

Example 1

The FTAC administrator wishes to view the admission profile PROFPROD with the command SHOW-FT-PROFILE to determine if the profile might endanger data protection. He uses the following command:

```
/SHOW-FT-PROFILE␣NAME=UMSAWARE,SELECT-PARAMETER=(OWNER-IDENTIFICATION
                                                 =DONALD),INFORMATION=*ALL
```

Short form:

```
/SHOW-FT-PROF␣PROFPROD,(,DONALD),*ALL
```

The output takes the following form:

```
%PROFPROD
% IGN-MAX-LEV = (IBR)
% FILE        = PROFIT
% USER-ADM    = (DONALD,M4711DON,OWN)
% PROC-ADM    = SAME
% DATA-ENC    = YES
```

The first line shows the name of the admission profile. The next two lines show the settings which Donald made in the command CREATE-FT-PROFILE using the parameters IGNORE-MAX-LEVELS=(INBOUND-RECEIVE=*YES) and FILE-NAME= PROFIT. The values for USER-ADMISSION and PROCESSING-ADMISSION have not been set by Donald, but rather the default values have been used. The output DATA-ENC=YES shows that requests are only accepted if the user data is encrypted. Donald established this using DATA-ENCRYPTION=*YES in the CREATE-FT-PROFILE command.

Example 2

The FTAC administrator examines the admission profile TESTPROF using the SHOW-FT-PROFILE command to determine whether file processing is possible with this profile. The command is as follows:

```
/SHOW-FT-PROFILE␣NAME=TESTPROF,SELECT-PARAMETER=(OWNER-IDENTIFICATION
                                                =DONALD),INFORMATION=*ALL
```

Abbreviated form:

```
/SHOW-FT-PROF␣TESTPROF,(,DONALD),INF=*ALL
```

The output has the following form:

```
%TESTPROF
% INITIATOR   = REMOTE
% USER-ADM    = (DONALD,*FIRST,OWN)
% PROC-ADM    = SAME
% FT-FUNCTION = (TRANSFER-FILE,FILE-PROCESSING)
```

The first line of the output displays the name of the admission profile. The second line indicates that the profile can only be used for requests initiated in the remote system. Donald has specified the value *FIRST for ACCOUNT in USER-ADMISSION; this means that the first account number assigned to the home pubset of the specified user ID in the system is used for account assignment in the case of transfer requests. As a result, it is unaffected by any changes to the account number. However, Donald has not specified a value for PROCESSING-ADMISSION and the default value SAME is therefore used. This means that the values are taken over from USER-ADMISSION. The last line indicates that the examined profile supports both file processing and file transfer requests

Command return codes

(SC2)	SC1	Maincode	Meaning
0	64	FTC0052	The information output was interrupted.
0	64	FTC0053	No FT profile exists which meets the specified criteria.
0	0	FTC0054	No information exists for the specified criteria.
0	64	FTC0153	The owner identification entered is not the user’s own ID.
0	64	FTC0171	The profile entered does not exist.
0	64	FTC0255	A system error occurred.

SC1/2 = Subcode 1/2 in decimal notation

For additional information, see section “Command return codes” on page 102

OPS variables

The following table shows the OPS variables of the SHOW-FT-PROFILE command with the operand INF=*ALL. The underlined values apply to the output with INF=*ONLY-NAMES.

Element	Type	Output
PROF-NAME	String	
PRIV	String	*YES / *NO
TRANS-ADM	String	*NSPEC / *SECRET
DUPLICATED	String	*YES / *NO
LOCKED-BY	Struct	
.IMPORT	String	*YES / *NO
.ADM	String	*YES / *NO
.USER	String	*YES / *NO
EXPIRED	String	*YES / *NO
USER-ADM	Struct	
.USER-ID	String	User-ID
.ACC	String	Account number / *FIRST / *NSPEC / *NRES
.PASSWORD	String	*OWN / *NSPEC / *NONE / *YES
EXP-DATE	String	yyyy-mm-dd / *NRES
USAGE	String	*PUBLIC / *PRIVATE / *NSPEC
IGNORE	Struct	
.OBS	String	*YES / *NO
.OBR	String	*YES / *NO
.IBS	String	*YES / *NO
.IBR	String	*YES / *NO
.IBP	String	*YES / *NO
.IBF	String	*YES / *NO
INITIATOR	String	*LOC / *REM / *NRES
TRANS-DIR	String	*FROM / *TO / *NRES
MAX-PART-LEV	String	Maximum security level / *NRES
PARTNERS	Array (1-50)	One or several partners / *NRES
FILE-NAME	String	File name / *NRES
LIBRARY	String	*YES / *NO / *NRES / Library
FILE-NAME-PREFIX	String	*YES / *NO

Element	Type	Output
ELEM	Struct	
.NAME	String	Name / *NRES / *NONE
.PREFIX	String	*YES / *NO
.VERSION	String	Version / *STD / *NONE / *NRES
.TYPE	String	Type / *NRES / *NONE
FILE-PASSWORD	String	*YES / *NRES / *NONE
WRITE	String	*NEW / *EXT / *REPL / *NRES
PROC-ADM	Struct	
.USER-ID	String	User-ID / *NRES / *SAME
.ACC	String	Account number / *NRES / *SAME
.PASSWORD	String	*NONE / *YES / *NRES / *SAME
SUCC	Struct	
.PROC	String	Commands / *NONE / *NRES / *EXPANSION
.PREFIX	String	Prefix / *NONE
.SUFFIX	String	Suffix / *NONE
FAIL	Struct	
.PROC	String	Commands / *NONE / *NRES / *EXPANSION
.PREFIX	String	Prefix / *NONE
.SUFFIX	String	Suffix / *NONE
TRANS-FILE	String	*ALLOWED / *NOT-ALLOWED
MOD-FILE-ATTR	String	*ALLOWED / *NOT-ALLOWED
READ-DIR	String	*ALLOWED / *NOT-ALLOWED
FILE-PROC	String	*ALLOWED / *NOT-ALLOWED
TEXT	String	Text / *NONE
DATA-ENC	String	*YES / *NO / *NRES

4.32 SHOW-FT-RANGE Display partner systems

Prerequisite for using this command is the use of *open*FT-AC.

The command SHOW-FT-RANGE is used to list the partner systems with which you can communicate by file transfer. In addition to indicating the name of the partner system, the security level is output which the FT administrator assigned to this system in the partner list. To determine which basic functions he is permitted to use, he must use the command SHOW-FT-ADMISSION-SET to obtain information on his admission set (see page 244).

The FTAC administrator can use SHOW-FT-RANGE to list all partner systems with which his FT system can communicate using file transfer. Furthermore, he can find out for any user in his system which partner systems can be accessed by this user.

```
SHOW-FT-RANGE

 USER-IDENTIFICATION = *OWN / <name 1..8>
,SELECT-PARAMETER = *ALL / *PARAMETERS(...)
    *PARAMETERS(...)
      | PARTNER = *ALL / <text 1..200 with-low>
 OUTPUT = *SYSOUT(...) / *SYSLST(...)
    *SYSOUT(...) / *SYSLST(...)
      | LAYOUT = *STD / *CSV
```

Operands

USER-IDENTIFICATION =
With USER-IDENTIFICATION, you can enter the user ID for which you would like to have a list of accessible partner systems.

USER-IDENTIFICATION = *OWN
This parameter can be used:
– by the FTAC user to list all the partner systems with which he can use at least one basic function.
– by the FTAC administrator to list all accessible partner systems.

USER-IDENTIFICATION = <name 1..8>
The FTAC user can only enter his own user ID here, the output is the same as the one for *OWN.
The FTAC administrator can enter any user ID for which he would like to view the accessible partner systems.

SELECT-PARAMETER =
With SELECT-PARAMETER, you can specify selection criteria for the partner systems.

SELECT-PARAMETER = *ALL
You wish to obtain information on all partner systems which can be reached.

SELECT-PARAMETER = *PARAMETERS(PARTNER = <text 1..200 with-low>)
You wish to obtain information on this partner system. You can specify the name from the partner list or the address of the partner system, see section “Specifying partner addresses” on page 39. Only two pieces of information are supplied:
– if you are permitted to communicate with this partner system
– the security level assigned to this partner system.

OUTPUT =
With OUTPUT, you can determine the output medium for the partner system listing.

OUTPUT = *SYSOUT(...)
The list is output on SYSOUT.

OUTPUT = *SYSLST(...)
The list is output on SYSLST.

LAYOUT = *STD
The output is put into a user-friendly form for reading.

LAYOUT = *CSV
The output is in **C**omma **S**eparated **V**alue format. This is a special tabular format, widely used in the PC world, where the individual fields are separated by a semicolon “;“ (see page 104 and 304).

Example

Donald Duck would like to find out about the security level of the computer BUYDUCK. To do this, he uses the following command:

```
/SHOW-FT-RANGE␣SELECT-PARAMETER=(PARTNER=BUYDUCK)
```

Short form:

```
/SHOW-FT-RANGE␣SEL=(BUYDUCK)
```

He receives the following output:

```
%SECLEV   PARTNER-NAME
%  50     BUYDUCK
```

The column SECLEV contains the security level of the partner system whose name appears in the PARTNER-NAME column. If Donald had entered SELECT-PARAMETER=*ALL (or left out this parameter altogether), he would have received a similar but longer list of all accessible partner systems.

Command return codes

(SC2)	SC1	Maincode	Meaning
0	64	FTC0052	The output of information was interrupted.
0	0	FTC0054	There is no information which meets the specified criteria.
0	64	FTC0070	The command cannot be executed on the basis of inadequate operating resources.
0	64	FTC0071	The FT subsystem has not yet been activated, the FTAC subsystem is therefore not active.
0	64	FTC0152	The user ID entered is not the user's own ID.
0	64	FTC0170	The partner entered is unknown within the partner systems possible for this user.
0	64	FTC0255	A system error occurred.

SC1/2 = Subcode 1/2 in decimal notation

For additional information, see section "Command return codes" on page 102

OPS variables

Element	Type	Output
SEC-LEV	Integer	Security level
PARTNER-NAME	String	Partner name

4.33 START-FT (FTSTART) Activate *open*FT

The START-FT command (or FTSTART) is used to activate the specified *open*FT instance. If you have not selected another *open*FT instance using SET-FT-INSTANCE, then start the standard instance.

The command is only executed if *open*FT is not active.

If the request queue contains file transfer requests for which the corresponding (remote) FT systems have also been started, these requests are started directly after *open*FT starts - provided the resources are available and no other start time has been defined.

It is possible to send SNMP traps and Console traps on START-FT.

Adequate steps must also be taken to ensure that all pubsets are available, as otherwise locally submitted requests that require unavailable pubsets are terminated with an error message. If this happens, the user cannot be notified by a result list or job variable.

If, in BCAM, the BCMAP FUNCT=INIT command of the MAXMAP parameter is used, the command must be unconditionally created before starting *open*FT.

If the *open*FT instance is to run under a virtual host name, the virtual host name must first be entered using MODIFY-FT-OPTIONS before the START-FT.

START-FT / FTSTART

Without operands

Correct execution of the START-FT command is acknowledged with the following message:

```
FTR0500 (&01'') openFT '(&00)' started. Protocols: '(&02)'
```

Command return codes

(SC2)	SC1	Maincode	Meaning
0	0	CMD0001	*open*FT system activated. The SYSOUT message contains the *open*FT version as an insert.
83	32	CMD0221	Internal error.
1	0	FTR1020	Command rejected. *open*FT already started
35	64	FTR1035	User not authorized for this command.
42	64	FTR1042	*open*FT could not be started.

SC1/2 = Subcode 1/2 in decimal notation

For additional information, see section “Command return codes” on page 102

4.34 START-OPENFTPART List partner systems as command procedure

The START-OPENFTPART command can be used to have all partner systems listed as a command procedure. MODIFY-FT-PARTNER commands are generated. This procedure can then be used to back up and maintain the partner list.

START-OPENFTPART
OUTPUT = <filename> **,PART**NER = ***ALL** / <text 1..8>

Operands

OUTPUT = <filename>
Specifies the name of the file to be created.

PARTNER = *ALL
All partner systems are included in the command procedure.

PARTNER = <text 1..8>
This is the name of the partner system (or partner systems) that is to be included in the command procedure.
This entry may be specified as a unique partner name (1 - 8 alphanumeric characters) or as a group of partners (1 - 7 characters, which must end with an asterisk “*”).

4.35 STOP-FT (FTSTOP) Deactivate *open*FT

The STOP-FT command (or FTSTOP) is used to initiate deactivation of the specified *open*FT instance and stop *open*FT.

The command is only executed if the instance has been started.

It is possible to send SNMP traps on STOP-FT.

STOP-FT / FTSTOP

Without Operands

Correct execution of the STOP-FT command is acknowledged with the following message:

```
%  FTR0501 openFT terminated
```

Command return codes

(SC2)	SC1	Maincode	Meaning
0	0	CMD0001	*open*FT system is terminated.
83	32	CMD0221	Internal error.
35	64	FTR1035	User not authorized for this command.
1	0	FTR1039	Command rejected. *open*FT is not active.

SC1/2 = Subcode 1/2 in decimal notation

For additional information, see section “Command return codes” on page 102

Example

Activate the local *open*FT system and subsequently deactivate the FT system:

```
/START-FT
FTR0500 (&01'') openFT '(&00)' started. Protocols: '(&02)'
.
.
.
/STOP-FT
FTR0501 openFT terminated
.
FTR0361 openFT control process terminated
```

4.36 UPDATE-FT-PUBLIC-KEYS (FTUPDKEY) Update public keys

Using this command, you can newly create the public key files of the key pair sets present in your *open*FT instance. This may become necessary if the existing public key files are unintentionally deleted. In addition, the command imports updated comments from SYSPKF.COMMENT to the public key files (see below).

The key pair consists of a private key, which is administered internally by *open*FT, and a public key. Public keys are stored on the configuration user ID of the *open*FT instance (default: $SYSFJAM) under the name **SYSPKF.R<key reference>.L<key length>**.

The key reference is a numeric designator for the version of the key pair. The key length is 768 by default. The public key files are text files that are created in the character code of the respective operating system, i.e. EBCDIC.DF04-1 for BS2000 and z/OS, ISO8859-1 for UNIX systems and CP1252 for Windows systems.

In a SYSPKF.COMMENT file on the configuration user ID of the *open*FT instance, you can store comments that are written in the first lines of this file when an existing public key file is updated. Such comments might contain, for example, the communications partner and the telephone number of the FT administrator on duty. The lines in the SYSPKF.COMMENT file may be a maximum of 78 characters in length.

Public key files with invalid key reference are automatically deleted (for example, public keys, for which *open*FT no longer has an internal private key).

UPDATE-**FT-PUB**LIC-**KEYS** / **FTUPDKEY**

Without operands

Command return codes

(SC2)	SC1	Maincode	Meaning
83	32	CMD0221	Internal error.
33	64	FTR1033	The public key files could not be updated.
35	64	FTR1035	The user is not authorized to use this command

SC1/2 = Subcode 1/2 in decimal notation

For additional information, see section “Command return codes” on page 102

5 Appendix

5.1 Structure of CSV outputs

5.1.1 Output format

The output format for all commands corresponds to the following rules:

– Each record is output in a separate line. A record contains all the information to be displayed on an object.

– The first line is a header and contains the field names of the respective columns. **Only the field names are guaranteed, not the order of fields in the record.** In other words, the order of columns is determined by the order of the field names in the header line.

– Two tables, with their own respective headers, are output sequentially for the commands SHOW-FILE-TRANSFER, SHOW-FTAC-ENVIRONMENT and SHOW-FT-PARTNERS. If one of the tables is empty, the corresponding header is also dropped.

– Individual fields within an output line are delimited by a semicolon “;”.

The following data types are differentiated in the output:

– Numeric

– String:
Since the “;” character in the CSV output is a metacharacter, text containing a “;” is enclosed within double quotes ("). Double quotes appearing within a text field are duplicated in order to differentiate them from text delimiters. When the text is imported into a program, all such duplicated quotes are eliminated, and all text delimiters are removed. Keywords are output in uppercase, starting with an asterisk (*), and are not enclosed within double quotes.

– Date:
The date and time are output in the form yyyy-mm-dd hh:mm:ss or only as yyyy-mm-dd.

5.1.2 SHOW-FILE-TRANSFER

Column	Type	Values	INF=
TransId	Numeric	Value	*STD/ *ALL
Initiator	String	*LOC / *REM	
State	String	*SUSP / *LOCK / *WAIT / *HOLD / *FIN / *ACT	
PartnerName	String	Value	
PartnerState	String	*ACT / *INACT / *NOCON / *INSTERR	
TransDir	String	*TO / *FROM	
ByteNum	Numeric	Value	
LocFileName	String	*NSPEC / Value	
LocElemName	String	*NSPEC / *NONE / Value	
LocElemType	String	*NSPEC / *NONE / Value	
LocElemVersion	String	*STD / *NONE /Value	
Prio	String	*LOW / *NORM / *HIGH	*ALL
Compress	String	*NONE / *BYTE / *ZIP	
DataEnc	String	*YES / *NO	
DiCheck	String	*YES / *NO	
Write	String	*REPL / *EXT / *NEW	
StartTime	yyyy-mm-dd hh:mm:ss	Value	
	String	*SOON / Value	
CancelTime	yyyy-mm-dd hh:mm:ss	Value	
	String	*NO	
Owner	String	Value	
DataType	String	*CHAR / *BIN / *NSPEC	
Transp	String	*YES / *NO	

Column	Type	Values	INF=
LocTransAdmId	String	*NONE / Value	*ALL
LocTransAdmAcc	String	*NONE / Value	
LocProfile	String	*NONE / Value	
LocProcAdmId	String	*NONE / Value	
LocProcAdmAcc	String	*NONE / Value	
LocSuccProc	String	*SECRET / *NONE / Value	
LocFailProc	String	*SECRET / *NONE / Value	
LocListing	String	*NONE / *SYSLST / *LISTFILE	
LocMonjv	String	*NONE / Value	
LocCcsn	String	*STD / Value	
RemFileName	String	*NSPEC / *NONE / Value	
RemElemName	String	*NSPEC / *NONE / Value	
RemElemType	String	*NSPEC / *NONE / Value	
RemElemVersion	String	*STD / *NONE / Value	
RemTransAdmId	String	*NONE / Value	
RemTransAdmAcc	String	Value / empty	
RemTransAdmAccount	String	Value / empty	
RemProfile	String	*YES / *NONE	
RemProcAdmId	String	*NONE / Value	
RemProcAdmAcc	String	*NONE / Value	
RemSuccProc	String	*SECRET / *NONE / Value	
RemFailProc	String	*SECRET / *NONE / Value	
RemCcsn	String	*STD / Value	
FileSize	Numeric	Value / empty	
RecSize	Numeric	Value / empty	
RecFormat	String	*STD / *VARIABLE / *FIX / *UNDEFINED	
StoreTime	yyy-mm-dd hh:mm:ss	Value	
ExpEndTime	yyy-mm-dd hh:mm:ss	Value / empty	
TranspMode	String	*YES / *NO	
DataEncrypt	String	*YES / *NO	
TabExp	String	*AUTO / *YES / *NO	
Mail	String	*ALL / *FAIL / *NO	

Column	Type	Values	INF=
DiagCode	String	Value / empty	*ALL
FileAvail	String	*IMMEDIATE / *DEFERRED / *NSPEC	
StorageAccount	String	Value / empty	
AccessRights	String	Value / empty	
LegalQualif	String	Value / empty	

Column	Type	Values	INF=
Act	Numeric	Value	*SUMMARY
Wait	Numeric	Value	
Lock	Numeric	Value	
Susp	Numeric	Value	
Hold	Numeric	Value	
Fin	Numeric	Value	
Total	Numeric	Value	

5.1.3 SHOW-FT-ADMISSION-SET

Column	Type	Values
UserID	String	Value
UserMaxObs	Numeric	Value
UserMaxObsStd	String	*YES / *NO
UserMaxObr	Numeric	Value
UserMaxObrStd	String	*YES / *NO
UserMaxIbs	Numeric	Value
UserMaxIbsStd	String	*YES / *NO
UserMaxIbr	Numeric	Value
UserMaxIbrStd	String	*YES / *NO
UserMaxIbp	Numeric	Value
UserMaxIbpStd	String	*YES / *NO
UserMaxIbf	Numeric	Value
UserMaxIbfStd	String	*YES / *NO
AdmMaxObs	Numeric	Value
AdmMaxObsStd	String	*YES / *NO
AdmMaxObr	Numeric	Value
AdmMaxObrStd	String	*YES / *NO
AdmMaxIbs	Numeric	Value
AdmMaxIbsStd	String	*YES / *NO
AdmMaxIbr	Numeric	Value
AdmMaxIbrStd	String	*YES / *NO
AdmMaxIbp	Numeric	Value
AdmMaxIbpStd	String	*YES / *NO
AdmMaxIbf	Numeric	Value
AdmMaxIbfStd	String	*YES / *NO
Priv	String	*YES / *NO
Password	String	*YES / *NO

5.1.4 SHOW-FT-LOGGING-RECORDS

Column	Type	Values	INF =
LogId	Numeric	Value	*STD / *ALL
ReasonCode	String	Value enclosed within double quotes to prevent interpretation as a number. FTAC Reason Codes are output as in the manual in the form of a Hex string (unlike OPS variables)	
LogTime	yyyy-mm-dd hh:mm:ss	Value	
InitUserId	String	Value / *REM	
InitTsn	String	Value / *NONE	
PartnerName	String	Value	
TransDir	String	*TO / *FROM / *NSPEC	
RecType	String	*FT / *FTAC	
Func	String	*TRANSFILE / *READ-FILE-ATTR / *DEL-FILE / *CRE-FILE / *MOD-FILE-ATTR / *READ-DIR / *MOVE-FILE / *CRE-FILE-DIR / *DEL-FILE-DIR / *MOD-FILE-DIR / *LOGIN	
UserAdmisId	String	Value	
FileName	String	Value	
Priv	String	*NO / *YES for FTAC logging records and with entry of an FTAC profile; otherwise, *NONE	
ProfName	String	Value / *NONE	
ResultProcess	String	*NONE / *STARTED / *NOT-STARTED	
StartTime	yyyy-mm-dd hh:mm:ss	Value	
	String	*NONE	
TransId	Numeric	Value	
	String	*NONE	
Write	String	*REPL / *EXT / *NEW / *NONE	
StoreTime	yyyy-mm-dd hh:mm:ss	Value	
	String	*NONE	

Column	Type	Values	INF =
ByteNum	Numeric	Value	
	String	*NONE	
DiagInf	String	*NONE / Value	
ErrInfo	String	Value / empty	
SecEncr	String	*YES / *NO	
SecDichk	String	*YES / *NO	*ALL
SecDencr	String	*YES / *NO	
SecDdichk	String	*YES / *NO	
SecLauth	String	*YES / *NO	
SecRauth	String	*YES / *NO	
RsaKeyLen	Numeric	Value	
SymEncrAlg	String	*DES / *AES	

5.1.5 SHOW-FT-OPTIONS

Column	Type	Values
PartnerLim	Numeric	Value
ReqLim	Numeric	Value
TaskLim	Numeric	Value
ConnLim	Numeric	Value
ReqWaitLev	Numeric	Value
TransportUnitSize	Numeric	Value
PartnerCheck	String	*STD / *TRANSP-ADDR
SecLev	Numeric	Value
TraceOpenft	String	*STD / *OFF
TraceOut [1]	String	*FILE / empty
TraceSession	String	*OFF
TraceFtam	String	*OFF / *STD
LogTransFile	String	*OFF / *ON
MaxInboundReq	Numeric	Value
MaxReqLifetime	String	*UNLIMITED / Value
SnmpTrapsSubsystemState	String	*OFF / *ON
SnmpTrapsFtState	String	*OFF / *ON
SnmpTrapsPartnerState	String	*OFF / *ON
SnmpTrapsPartnerUnreach	String	*OFF / *ON
SnmpTrapsReqQueueState	String	*OFF / *ON
SnmpTrapsTransSucc	String	*OFF / *ON
SnmpTrapsTransFail	String	*OFF / *ON
ConsoleTraps	String	*OFF / *ON
TeleService	String	FT / FTAC / FT/FTAC
HostName	String	Value
Identification	String	Value
UseTns	String	*YES / *NO
ConsTrapsSubsystemState	String	*ON / *OFF
ConsTrapsFtState	String	*ON / *OFF
ConsTrapsPartnerState	String	*ON / *OFF
ConsTrapsPartnerUnreach	String	*ON / *OFF
ConsTrapsReqQueueState	String	*ON / *OFF

Column	Type	Values
ConsTrapsTransSucc	String	*ON / *OFF
ConsTrapsTransFail	String	*ON / *OFF
FtLog	String	*ALL / *NONE / *FAIL
FtacLog	String	*ALL / *NONE / *FAIL / *MODIFY
Trace	String	*ON / *OFF
TraceSelp	String	*ALL / Value
TraceSelr	String	*ALL / Value
TraceOpt	String	*NONE / Value
KeyLen	Numeric	Value
CcsName	String	Value
AppEntTitle	String	*YES / *NO
StatName	String	Value
SysName	String	Value / empty
FtStarted	String	*YES / *NO
openftAppl	String / Numeric	*STD / Value
ftamAppl	String /Numeric	*STD / Value
FtpPort	String	Value / empty
ftstdPort	Numeric	Value
DynPartner	String	*ON / *OFF

[1] TraceOut only applies with *open*FT partners

5.1.6 SHOW-FT-PARTNERS

Column	Type	Values
PartnerName	String	Value
Sta	String	*ACT / *DEACT / *NOCON / *LUNK / *RUNK / *ADEAC / *AINACT / *LAUTH / *RAUTH / *NOKEY / *DIERR / *IDREJ
SecLev	String	*STD / *B-P-ATTR
	Numeric	Value
Trace	String	*FTOPT / *ON / *OFF
Loc	Numeric	Value
Rem	Numeric	Value
Processor	String	Value
Entity	String	Value
NetworkAddr	String	Value
Port	Integer	Value
PartnerCheck	String	*FTOPT / *STD / *TRANSP-ADDR / *AUTH / *AUTHM / *NOKEY
TransportSel	String	Value
LastAccessDate	yyyy-mm-dd	Value
NetworkAddr	String	Value
TransportSel	String	Value
SessionSel	String	Value / *NONE
PresentationSel	String	Value / *NONE
Identification	String	Value
SessRout	String	Value / *ID
PartnerAddr	String	Value
Check	String	*FTOPT / *STD / *TRANSP-ADDR
AuthMand	String	*YES / *NO

5.1.7 SHOW-FT-PROFILE

Column	Type	Values	INF=
ProfName	String	Value	*ONLY-NAMES / *ALL
Priv	String	*YES / *NO	
TransAdm	String	*NSPEC / *SECRET	
Duplicated	String	*YES / *NO	
LockedByImport	String	*YES / *NO	
LockedByAdm	String	*YES / *NO	
LockedByUser	String	*YES / *NO	
Expired	String	*YES / *NO	
UserAdmId	String	Value	
UserAdmAcc	String	Value / *FIRST/*NSPEC / *NRES	*ALL
UserAdmPass	String	*OWN / *NSPEC / *NONE / *YES	
ExpDate	yyyy-mm-dd	Value / *NRES	
Usage	String	*PUBLIC / *PRIVATE / *NSPEC	
IgnObs	String	*YES / *NO	
IgnObr	String	*YES / *NO	
IgnIbs	String	*YES / *NO	
IgnIbr	String	*YES / *NO	
IgnIbp	String	*YES / *NO	
IgnIbf	String	*YES / *NO	
Initiator	String	*LOC / *REM / *NRES	
TransDir	String	*FROM / *TO / *NRES	
MaxPartLev	Numeric	Value	
	String	*NRES	
Partners	String	One or more FT partners, separated by commas / *NRES	
FileName	String	Value / *NRES	
Library	String	*YES / *NO / *NRES / Value	
FileNamePrefix	String	*YES / *NO	
ElemName	String	Value / *NRES / *NONE	
ElemPrefix	String	*YES / *NO	
ElemVersion	String	Value / *STD / *NONE	
ElemType	String	Value / *NRES / *NONE	

Column	Type	Values	INF=
FilePass	String	*YES / *NRES / *NONE	*ALL
Write	String	*NEW / *EXT / *REPL / *NRES	
ProcAdmId	String	Value / *NRES / *SAME	
ProcAdmAcc	String	Value / *NRES / *SAME	
ProcAdmPass	String	*NONE / *YES / *NRES / *SAME	
SuccProc	String	Value / *NONE / *NRES / *EXPANSION	
SuccPrefix	String	Value / *NONE	
SuccSuffix	String	Value / *NONE	
FailProc	String	Value / *NONE / *NRES / *EXPANSION	
FailPrefix	String	Value / *NONE	
FailSuffix	String	Value / *NONE	
TransFile	String	*ALLOWED / *NOT-ALLOWED	
ModFileAttr	String	*ALLOWED / *NOT-ALLOWED	
ReadDir	String	*ALLOWED / *NOT-ALLOWED	
Text	String	Value / *NONE	
FileProc	String	*ALLOWED / *NOT-ALLOWED	
DataEnc	String	*YES / *NO / *NRES	

5.1.8 SHOW-FT-RANGE

Column	Type	Value
SecLev	Numeric	Degree of security level
PartnerName	String	Partner name

5.1.9 SHOW-FTAC-ENVIRONMENT

The command SHOW-FTAC-ENVIRONMENT sequentially displays the objects contained in an FTAC export file in a format that corresponds to the output of the SHOW-FT-ADMISSION-SET(page 297) and SHOW-FT-PROFILE (page 303) commands.

5.2 Accounting records

Structure of the FT accounting records

The FT accounting record is divided into four parts:

1. record definition
2. identification section
3. basic information
4. variable information

The record sections contain the displacement, length and format of the data field.

The **field number** identifies the sequence number within the part of the record written.

The **displacement** is the position of the data field relative to the beginning of the part of the record that has been written.

The **length** is the length of the data field in bytes.

The **format** is the format of the data field:

- A = alphanumeric (including $, #, and @)
- B = binary number
- C = printable characters
- F = file name for BS2000
- Z = unpacked decimal number (0...9)
- – = undefined

1. Record definition section

The record definition section contains the record identifier, the time of day, the length of the identification section and the length of the basic information.

Field No.	Displ.	Length	Format	Meaning
1	00	4	A	Record identifier ‘FTR0‘
2	04	8	- /B	Time stamp of the time-of-day clock
3	0C	2	B	Length of the identification section
4	0E	2	B	Length of the basic information
5	10	4	-	Reserved

Layout of the record definition section

2. Identification section

The identification section contains the user ID, account number and job number (TSN).

Field No.	Displ.	Length	Format	Meaning
1	00	8	A	User ID
2	08	8	A	Account number
3	10	4	Z	Job number (TSN) (This field applies only to locally issued requests.)

Layout of the identification section

3. Basic information

The basic information includes

– Date and time when the FT request was stored,

– Date and time when the transfer ended,

– Result of the transfer,

– Details of the start of follow-up processing,

– Name of the remote system,

– Indication as to whether the accounting record was written by the local or the remote system,

– Identification of the FT request,

– Number of disk accesses,

– Number of bytes written to or read from disk,

– Number of bytes sent to or read from the accounting record.

Field No.	Displ.	Length	Format	Meaning
1	00	12	Z	Time when the file transfer request was stored (format: yymmddhhmmss; this field applies only to locally issued requests)
2	0C	12	Z	Time when the transfer ended (format: yymmddhhmmss)
3	18	1	C	Result of the transfer: + : successful execution - : execution with errors 0 : not used
4	19	1	C	Result of the start of follow-up: + : successful execution - : execution with errors 0 : not used
5	1A	8	A	Partner name
6	22	1	A	Specifies whether the request was issued in local or remote system: L : the request was submitted in the local system R : the request was submitted in the remote system
7	23	11	Z	Transfer ID
8	2E	2	-	Reserved
9	30	4	-	Reserved

Layout of the basic information

Field No.	Displ.	Length	Format	Meaning
10	34	4	B	Number of disk accesses
11	38	8	B	Number of bytes on disk
12	40	8	B	Number of bytes in network

Layout of the basic information

4. Variable information

Field No.	Displ.	Length	Format	Meaning
1	00	2	B	Number of extensions = 4
2	02	2	B	Displacement of the record extension for the file name from the start of record
3	04	2	B	Displacement of the record extension for the library member name from the start of record
4	06	2	B	Displacement of the record extension for the century part of the time specification from the start of the record
5	08	2	B	Displacement of the record extension for the CPU time from the start of the record
The variable information includes the file name and the name of the library member.				

Header of the variable section

Field No.	Displ.	Length	Format	Meaning
1	00	2	A	Extension identification = ‘FN‘
2	02	1	B	Extension type = x‘00‘
3	03	1	B	Length of the file name
4	04	see field 3	F	File name
If a displacement is set to 0, the corresponding record extension has not been specified.				

Record extension for the file name

Field No.	Displ.	Length	Format	Meaning
1	00	2	A	Extension identification = ‘MN‘
2	02	1	B	Extension type = x‘00‘
3	03	1	B	Length of extension (not including identification, type and length field)
4	04	8	A	Library member type
5	0C	24	A	Library member version
6	24	8	Z	Library member variant
7	2C	1	B	Length of library member name
8	2D	see field 7	A	Library member name

Record extension for the library member name

Field No.	Displ.	Length	Format	Meaning
1	00	2	A	Extension identification = ‘YY‘
2	02	1	B	Extension type = x‘00‘
3	03	1	B	Length of extension (not including identification, type, length field) = 4
4	04	2	Z	Time at which the request was stored in the form yy (see field 1 in the basic information)
5	06	2	Z	Time of transfer in the form yy (see field 2 in basic information)

Record extension for the century part of the time specification

Field No.	Displ.	Length	Format	Meaning
1	00	2	A	Extension identification = ‘MS‘
2	02	1	B	Extension type = x‘00‘
3	03	1	B	Length of extension (not including identification, type, length field) = 4
4	04	4	B	Number of machine commands required in the local system by this request (in units of 10,000 commands)

Record extension for CPU time

5.3 Recovering from hung FT and FTAC subsystems

In some cases, e.g. when system errors occur, it may not be possible to unload the subsystems. This may be due to the following reasons:

– The subsystem is in a LOCKED state, since the associated holder task can no longer be used, e.g. after a system dump.

– The subsystem cannot be unloaded because some tasks are still connected. This typically occurs when FT cannot be exited (the tasks with the TSNs FTC* and the FT server tasks do not disappear) when connected tasks enter permanent wait states, or when tasks are “permanently pended” after system dumps due to insufficient disk space.

In such exceptional cases, the system administrator can resort to some special resources to unload the subsystems and thus save the BS2000 session. These resources are described in the DSSM manual. The following points discuss some of the aspects to be observed when unloading the FT subsystems.

WARNING!

There is always a certain risk involved in using any such resources. There is essentially no way of guaranteeing that all error states are fully described here. The ultimate responsibility always lies with the system administrator !

A subsystem in a LOCKED state can be removed from the system by using the command /UNLOCK-SUBSYSTEM. Note, however, that this does **not** call the subsystem-specific uninstallation routine and therefore has the following consequences for the FT subsystem:

- No file locks held by FT are released, so all user files locked by FT will remain locked. These locks can be removed explicitly by the system administrator with VERIFY or will disappear implicitly at the next BS2000 startup.
- The request logs of all started *open*FT instances (SYSFSF on the configuration user IDs) also remain locked. They must be unlocked using VERIFY before the FT subsystem is started again.

A subsystem to which tasks are still connected can be unloaded with /STOP-SUBSYSTEM FORCE=YES if required, provided the attribute FORCED=ALLOWED is first assigned with the command /MOD-SUBSYSTEM-PARAMETERS. By default, FT subsystems do not have this attribute.

This approach causes any restarted tasks that are still connected to run with a system dump. Such system dumps are of no use whatsoever for any diagnostic purposes and may hence be discarded. FT tasks which are still connected and which are stuck in a bourse will run within at most 10 minutes of the system dump.

The subsystem should never be restarted as long as there are existing tasks which were still connected at the time of forcing the subsystems to unload!

Glossary

Below you will find short definitions of some of the technical terms used in this manual. *Italic* type indicates a reference to other terms in this glossary.

access control
: *File attribute* in the *virtual filestore*, attribute of the *security group* that defines *access rights*.

access protection
: Comprises all the methods used to protect a data processing system against unauthorized system access.

access right / access admission
: Derived from the *transfer admission*. It defines the scope of access for the user who specifies the transfer admission.

ACL (Access Control List)
: -> see *Access Control List*

action list
: Component of the file attribute *access control* (attribute of the *security group*) in the *virtual filestore* that defines *access rights*.

admission profile
: Way of defining the *FTAC* protection functions. Admission profiles define a *transfer admission* that has to be specified in *FT requests* instead of the *LOGON or Login authorization*. The admission profile defines the *access rights* for a user ID by restricting the use of parameters in *FT requests*.

admission profile, privileged
: -> see *privileged admission profile*

admission set
: In *FTAC*, the admission set for a particular user ID defines which FT functions the user ID may use and for which *partner systems*.

admission set, privileged

-> see *privileged admission set*

AES (Advanced Encryption Standard)

The current symmetrical encryption standard, established by NIST (National Institute of Standards and Technology), based on the Rijndael algorithm, developed at the University of Leuven (B).

alphanumeric

Alphanumeric characters comprise alphabetic and numeric characters, i.e. the letters A-Z and the digits 0-9.

ANSI code

Standardized 8-bit character code for message exchange. The acronym stands for "American National Standards Institute".

API (Application Program Interface)

An interface that is freely available to application programmers. It provides a set of interface mechanisms designed to support specific functionalities.

Application Entity Title (AET)

The Application Entity Title consists of Layer 7 addressing information of the *OSI Reference Model*. It is only significant for *FTAM partners*.

ASECO (Advanced Security Control)

-> see *Advanced Security Control*

asynchronous request

Once the *FT request* has been submitted, it is processed independently of the user. The user can continue working once the system has confirmed acceptance of the request. (see also *synchronous request*).

audit

Fundamental function of a secure system; logging of operating sequences and preparation of the logged data.

authentication

– Security service that validates a predefined identity.
– Process used by *open*FT to check the unique identity of the request partner.

character repertoire

Character set of a file in the *virtual filestore.*
In the case of files transferred with *FTAM partners* it is possible to choose between: GeneralString, GraphicString, IA5String and VisibleString.

client

- Term derived from client/server architectures: the partner that makes use of the services provided by a *server*.
- Logical instance which submits requests to a *server*.

Comma Separated Value (CSV)

This is a quasi-tabular output format that is very widely used in the PC environment in which the individual fields are separated by a semicolon ";". It permits the further processing of the output from the most important *open*FT commands using separate tools.

communication computer

Computer for constructing a *data communication system*.

communication controller

-> see *preprocessor*

compression

This means that several identical successive characters can be reduced to one character and the number of characters is added to this. This reduces transfer times.

computer network, open

-> see *open computer network*

concurrency control

Component of the FTAM file attribute *access control* (part of the *security group*) in the *virtual filestore* that controls concurrent access. *open*FT for BS2000 offers only passive and partial support for concurrency control. Note: "partial support" is a technical term taken from the FTAM environment that means that the parameter is interpreted correctly at the syntactic level but is not genuinely supported.

configuration user ID

Each *open*FT instance in BS2000 requires an ID, on which the variable files of this file are stored (for the default instance: $SYSFJAM).

connectivity

In general, the ability of systems and partners to communicate with one another. Sometimes refers simply to the communication possibilities between transport systems.

constraint set

Component of the *document type*.

contents type

File attribute in the *virtual filestore,* attribute of the *kernel group* that describes the file structure and the form of the file contents.

cross domain connection

Connection mode in which a TRANSDATA network is connected as an SNA domain to an SNA domain via a *gateway.*

data communication system

Sum of the hardware and software mechanisms which allow two or more communication partners to exchange data while adhering to specific rules.

data compression

Reducing the amount of data by means of compressed representation.

data encoding

Way in which an *FT system* represents characters internally.

Data Encryption Standard (DES)

International data encryption standard for improved security. The DES procedure is used in the FT products of Fujitsu Siemens Computers to encrypt the request description data and possibly the request data.

data protection

- In the narrow sense as laid down by law, the task of protecting personal data against misuse during processing in order to prevent the disclosure or misappropriation of personal information.
- In the wider sense, the task of protecting data throughout the various stages of processing in order to prevent the disclosure or misappropriation of information relating to oneself or third parties.

data security

Technical and organizational task responsible for guaranteeing the security of data stores and data processing sequences, intended in particular to ensure that

- only authorized personnel can access the data,
- no undesired or unauthorized processing of the data is performed,
- the data is not tampered with during processing,
- the data is reproducible.

directory

In the tree-like UNIX file system or BS2000 (POSIX), directories exist in the form of lists that refer to files and to other directories.
In BS2000 (DVS), PLAM libraries are interpreted as directories.

document type

Value of the file attribute *contents type* (attribute of the *kernel group*). Describes the type of file contents in the *virtual filestore*.

– *document type* for text files: FTAM-1
– *document type* for binary files: FTAM-3

emulation

Components that mimic the properties of another device.

entity

-> see *instance*

Explorer

A program from Microsoft that is supplied with Windows operating systems to facilitate navigation within the file system.

file attributes

A file's properties, for example the size of the file, access rights to the file or the file's record structure.

file directory / file catalog

File present in every *pubset* (in *SM pubsets* there is a file directory in every volume set). All a pubset's files and job variables are entered in the corresponding *file directory*. Files on private disks and tapes can be entered in the file directory. A catalog entry contains all a file's or job variable's attributes (protection attributes, location of the administered data etc.

file management

Possibility of managing files in the remote system. The following actions are possible:

– Create directories
– Display and modify directories
– Delete directories
– Display and modify file attributes
– Rename files
– Delete files.

file processing

The *openFT* "file processing" function makes it possible to send a receive request in which the output of a remote command or program is transferred instead of a remote file.

filestore, virtual

-> see *virtual filestore*

file transfer request

-> see *FT- request*

fixed-length record

A record in a file all of whose records possess the same, agreed length. It is not necessary to indicate this length within the file.

FJAM-LU

FT-specific software module that is required for the connection of *open*FT for z/OS to a *TRANSDATA network* via *TRANSIT-SNA*. FJAM-LU is a component of TRANSIT-SNA.

follow-up processing

FT function that initiates execution of user-specified commands or statements in the *local* and/or the *remote system* after an *FT request* has been completed. The user may define different follow-up processing, depending on the success or failure of FT request processing. See also *preprocessing* and *postprocessing*.

follow-up processing request

Statements contained within an *FT request* for *follow-up processing* to be performed after file transfer.

FTAC (File Transfer Access Control)

Extended access control for file transfer and file management. In the case of BS2000 and z/OS, this is implemented by means of the product *open*FT-AC, for other operating systems it is a component of the *open*FT product, e.g. in *open*FT for UNIX systems.

FTAC administrator

Person who manages *open*FT-AC on a computer, i.e. who is responsible, among other things, for the entries in the *network description file* as well as for controlling resources.

FTAC-BS2000 / FTAC-SINIX

Original name of the add-on products offering extended access protection. Now known under the name *open*FT-AC.

FT administrator

Person who administers the *open*FT product installed on a computer, i.e. who is responsible, among other things, for the entries in the *network description file* as well as for controlling resources.

FTAM file attributes

All systems which permit file transfer via FTAM protocols must make their files available to their partners using a standardized description (ISO 8571). To this end, the attributes of a file are mapped from the physical filestore to a *virtual filestore* and vice versa. This process distinguishes between three groups of file attributes:

– kernel group: describes the most important file attributes.
– storage group: contains the file's storage attributes.
– security group: defines security attributes for file and system access control.

FTAM partner

Partner system that uses *FTAM protocols* for communication.

FTAM protocol (File Transfer, Access and Management)

Protocol for file transfer standardized by the "International Organization for Standardization" (ISO) (ISO 8571, FTAM).

FTIF (File Transfer Interconnect Facility)

Has the task of interconnecting different transport systems for file transfer; implemented in *open*FTIF for UNIX systems or Windows.

FTIF gateway

Computer on which *open*FTIF is installed.

FTIF name

Name used by *open*FTIF to identify the partner application in the destination system. This name is specified as a symbolic name (also referred to as GLOBAL NAME) for the partner application in a TNS entry in the FTIF gateway.

FTP partner

Partner system that uses *FTAM protocols* for communication.

FTP protocol

Manufacturer-independent protocol for file transfer in TCP/IP networks.

FT request

Request to an *FT system* to transfer a file from a *send system* to a *receive system* and (optionally) start *follow-up processing requests*.

FT system

System for transferring files that consists of a computer and the software required for file transfer.

FT trace

Diagnostic function that logs FT operation.

functionality class

Class which places certain minimum security function demands on an IT system.
The functionality classes are defined in the “Kriterien für die Bewertung der Sicherheit von Systemen der Informationstechnik (IT)”, (Criteria for the assessment of the security of Onformation Technology (IT) systems), version 1 of 11th January 1989, published by the Zentralstelle für Sicherheit in der Informationstechnik (Central Office for Security in Information technology) on behalf of the German government.

functional standard

Recommendation defining the conditions and the forms of application for specific ISO standards (equivalent term: *profile*). The transfer of unstructured files is defined in the European Prestandard CEN/CENELEC ENV 41 204; file management is defined in the European Prestandard CEN/CENELEC ENV 41205.

gateway

Generally understood to mean a computer that connects two or more networks and which does not function as a bridge. Variants: gateway at network level (=router or OSI relay), transport and application gateway.

gateway processor

Communication computer that links a computer network to another *computer network*. The mapping of the different protocols of the various *computer networks* takes place in gateway processors.

global privileges

All the privileges that can be assigned using the /SET-PRIVILEGE command including the security administrator privilege and the TSOS privilege. *Global privileges and system administrator privileges* are identical.

global user administration

This comprises the administration of user IDs and user groups and covers resources and user rights, the creation, modification and deletion of user IDs and user groups

guard

A component of the GUARDS condition administration system. A guard unites conditions which are evaluated by the standard GUARDS condition administration system on request.

GUARDS (Generally Usable Access Control Administration System)
Object administration for *Guards*.

heterogeneous network
A network consisting of multiple subnetworks functioning on the basis of different technical principles.

homogeneous network
A network constructed on the basis of a single technical principle.

host
Formerly a large-scale data processing system which required a *front-end processor* in order to be able to communicate. Nowadays, the term used for BS2000 or z/OS systems.

identification
Procedure making it possible to identify a person or object.

inbound request / inbound submission
Request issued in another system, i.e. for this request.

initiator
Here: *FT system* that submits an *FT request*.

instance / entity
A concept of OSI architecture: active element in a layer. Also see *openFT instance.*

instance ID
A network-wide, unique address of an *open*FT instance.

integrity
Unfalsified, correct data following the processing, transfer and storage phases.

Internet
Communications architecture characterized by the use of TCP/IP, evolved from the ARPANET in the USA. Extensions are monitored by the IAB using the *RFC* process.

interoperability
Capability of two *FT systems* to work together.

ISO/OSI reference model
The ISO/OSI Reference Model is a framework for the standardization of communications between open systems. (ISO=International Standards Organization).

job
Sequence of commands, statements and data.

job class
Job classes combine *jobs* which share certain properties and characteristics.

job transfer
Transfer of a file that constitutes a *job* in the *receive system* and is initiated as a job there.

joinfile / user catalog / user ID catalog
File that contains the *user attributes* of all the *user IDs* in a *pubset*.

kernel group
Group of file attributes of the *virtual filestore* that encompasses the kernel attributes of a file.

library
File with internal structure (members)

library member
Part of a library. A library member may in turn be subdivided into a number of records.

Local Area Network (LAN)
Originally a high-speed network with limited physical extension. Nowadays, any network, that uses CSMA/CD, Token Ring or FDDI irrespective of the range (see also *WAN Wide Area Network*).

local system
The *FT system* at which the user is working.

logging function
Function used by *openFT* to log all file transfer accesses to the protected system.

logging record
Contains information about access checks performed by *openFT*.

Logical Unit (LU)
Interface between an application program and the SNA data communications network. The LU type describes the communications characteristics.

Login authorization
Transfer admission to a computer which (as a rule) consists of the login name and the password, and authorizes dialog operation, see also *LOGON authorization*.

LOGON authorization
Transfer admission authorizing access to a computer. The LOGON authorization (normally) consists of user ID, account number and password and authorizes the user to make use of interactive operation.

mainframe
Computer (consisting of one or more processors) which runs under the control of a universal operating system (e.g. BS2000).
Synonyms: BS2000 computer, host computer.

maximum-string-length
Specifies the maximum length of *strings* within a file in the *virtual FTAM filestore*.

Network Control Program (NCP)
Operating system of the front-end-processor for SNA hosts.

NEA
Name of a network architecture.

network description file
File used up to *open*FT V9 that contains specifications concerning *remote systems* (*FT systems*).

Network Management Kernel
Component of the Network Management Platform; responsible for forwarding network management requests as well as for centralized tasks such as logging, authorization checks, request and application administration.

object
Passive element in a DP system that contains or receives data and which can be the object of an operation such as read, write or execute etc.
Examples: files, user IDs

open computer network
: Computer network in which communication is governed by the rules of ISO/OSI. Interoperation of different computers from various vendors is made possible by defined *protocols*.

***open*FT-FTAM**
: Add-on product for *open*FT (for BS2000, UNIX systems and Windows systems) that supports file transfer using FTAM protocols. FT-FTAM stands for File Transfer FTAM support.

***open*FTIF**
: *open*FTIF performs the task of interconnecting different transport systems for file transfer.

***open*FT instance**
: Several *open*FT systems, so-called *open*FT instances, can be running simultaneously on on the HIPLEX cluster . Each instance has its own address (instance ID, virtual BCAM host) and is comprised of the loaded code of the *open*FT products (including add-on products if they are available) and of the variable files such as the network description file, logging files, key library, request log, etc.

***open*FT partner**
: *Partner system* which is communicated with using *openFT protocols*.

***open*FT protocols**
: *Protocols* standardized by Siemens AG for file transfer (SN77309, SN77312).

operating parameters
: Parameters that control the *resources* (e.g. the permissible number of connections).

outbound request / outbound submission
: Request issued in your own processor.

owner of an FT request
: User ID in the *local system* or *remote system* under which the *FT request* is started (or submitted). The owner is always the ID under which the request is submitted, not the ID under which it is executed.

partner list
: File containing specifications concerning *remote systems* (*FT systems*).

partner system

Here: *FT system* that carries out *FT requests* in cooperation with the *local system*.

password

Sequence of characters that a user must enter in order to access a user ID, file, job variable, network node or application. The user ID password serves for user *authentication*. It is used for access control. The file password is used to check access rights when users access a file (or job variable). It is used for file protection purposes.

PDN

Communication computer control program, consisting of the computer's operating system and system programs for the handling of communications protocols. Software that runs on a TRANSDATA data communications computer.

permitted actions

File attribute in the *virtual filestore*; attribute of the *kernel group* that defines actions that are permitted in principle.

Personal Audit for Individual Accountability

Trace of individual system utilization. Identification can take the following forms:

– a user ID corresponds to a user, or
– a user may use only one operator terminal.

Physical Unit (PU)

port number

Number that uniquely identifies a TCP/IP application or the end point of a TCP/IP connection within a processor.

POSIX (Portable Open System Interface)

Board and standards laid down by it for interfaces that can be ported to different system platforms.

postprocessing

*open*FT makes it possible to process the received data in the receiving system through a series of operating system commands, under the process control of *open*FT (in contrast to *follow-up processing*).

preprocessing

The preprocessing facility in *open*FT can be used to send a receive request in which the outputs of a remote command or program are transferred instead of a file. This makes it possible to query a database on a remote system, for example. Preprocessing also may be issued locally.

preprocessor / communication controller

A processor system connected upstream of the mainframe which performs special communication tasks in the network. Synonym: communication processor.

presentation

Entity that implements the presentation layer (layer 6) of the *ISO/OSI Reference Model* in an *FT system* that uses e.g. *FTAM protocols*.

presentation selector

Subaddress used to address a *presentation application*.

private key

Secret decryption key used by the recipient to decrypt a message that was encrypted using a *public key*. Used by a variety of encryption procedures including the *RSA procedure*.

privilege

– Global privilege within the system that authorizes a user to execute certain commands and call certain program interfaces (e.g. TSOS privilege).
– Set of user-specific attributes that are used by the access control system.

privileged admission profile

Admission profile that allows the user to exceed the *FTAC administrator's* presettings in the *admission set*. This must be approved by the FTAC administrator who is the only person able to privilege admission profiles.

privileged admission set

Admission set belonging to the *FTAC administrator*.

processor node

Entity in the host or communication computer that can be addressed throughout the network and that performs service functions for the exchange of data.

profile

In OSI, a profile is a standard which defines which protocols may be used for any given purpose and specifies the required values of parameters and options. Here: a set of commands assigned to a user ID. The permissibility of these commands is ensured by means of syntax files. See also *admission profile*, *privileged admission profile*.

protocol

Set of rules governing information exchange between peer partners in order to achieve a defined objective. This usually consists of a definition of the messages that are to be exchanged and the correct sequencing of messages including the handling of errors and other exceptions.

public key

Public encryption key defined by the receiver of a message, and made public or made known to the sender of the message. This allows the sender to encrypt messages to be sent to the receiver. Public keys are used by various encryption methods, including the *Rivest Shamir Adleman (RSA) procedure*. The public key must match the *secret key* known only to the receiver.

public space

Named disk storage area which is available to a defined number of user IDs within the operating system. This storage area may be located on one or more Public Volume Sets (*pubsets*).

pubset / public volume set

Set of shared, named disk storage units which is defined by a catalog identification (catid). A distinction is made between *SF pubsets* and *SM pubsets*.

receive file

File in the *receive system* in which the data from the *send file* is stored.

receive system

System to which a file is sent. This may be the *local system* or the *remote system*.

record

Set of data that is treated as a single logical unit.

relay

OSI term for an element in a layer that acts as an intermediary between two other partners and thus makes communications between these two partners possible.
In the narrow sense, on the network layer a relay is the functional equivalent of a *router*.

relay program
Program in a *gateway processor* that maps the different protocols onto one another (e.g. TRANSIT products).

remote system
-> see *partner system*

request
Here: *FT request,* in BS2000: *Job*

request class / job class
-> see *job class*

request file
File containing *asynchronous requests* and their processing statuses.

request identification / request ID / request number
The (serial) number assigned to the request by the system. In some commands, users are able to identify the request on the basis of this number.
Here: number that identifies an *FT request.*

request management
FT function responsible for managing *FT requests*; it ensures request processing from the submission of a request until its complete processing or termination.

request number
-> see *request identification*

request storage
FT function responsible for storing *FT requests* until they have been fully processed or terminated.

resources
Hardware and software components needed by the *FT system* to execute an *FT request* (*tasks*, connections, lines). These resources are controlled by the *operating parameters*.

responder
Here: *FT system* addressed by the *initiator*.

restart
Automatic continuation of an *FT request* following an interruption.

restart point

Point up to which the data of the *send file* has been stored in the *receive file* when a file transfer is interrupted and at which the transfer of data is resumed following a *restart*.

result list

List with information on a completed file transfer. This is supplied to the user in the *local system* and contains information on his or her *FT requests*.

RFC (Request for Comments)

Procedure used on the Internet for commenting on proposed standards, definitions or reports. Also used to designate a document approved in this way.

RFC1006

Supplementary protocol for the implementation of ISO transport services (transport class 0) using TCP/IP.

Rivest-Shamir-Adleman-procedure (RSA procedure)

Encryption procedure named after its inventors that operates with a key pair consisting of a *public key* and a *private key*. Used by FT products in order to reliably check the identity of the partner system and to transmit the AES key to the partner system for encrypting the request data.

router

Network element that is located between networks and guides message flows through the networks while simultaneously performing route selection, addressing and other functions. Operates on layer 3 of the OSI model.

RPC (Remote Procedure Call)

Cross-network server procedure call issued by client.

security attributes

An object's security attributes specify how and in what ways the object may be accessed.

security group

Group of file attributes in the *virtual filestore*, encompassing the security attributes of a file.

security level

When FTAC is used, the security level indicates the required level of protection against a *partner system*.

send file

File in the *send system* from which data is transferred to the *receive file*.

send system

Here: *FT system* that sends a file. This may be the *local system* or the *remote system*.

server

Logical entity or application component which executes a client's requests and assures the (coordinated) usage of all the generally available services (File, Print, DB, Communication, etc.). May itself be the client of another server.

service

– As used in the OSI architecture: a service is the set of functions that a service provider makes available at a service access point.
– As used in the client/server architecture: a set of functions that a server makes available to its clients.

service class

Parameter used by *FTAM partners* to negotiate the functions to be used.

session

– In OSI, the term used for a layer 5 connection.
– In SNA, a general term for a connection between communication partners (applications, devices or users).

session selector

Subaddress used to address a *session* application.

SF pubset (Single Feature Pubset)

One or more disks whose key properties (disk format, allocation unit) match and which are used to store files and JVs under a shared catalog ID.

SNA network

Data communication system that implements the Systems Network Architecture (SNA) of IBM.

SNMP (Simple Network Management Protocol)

Protocol for TCP/IP networks defined by the Internet Engineering Task Force (IETF) for the transfer of management information.

Standard Access Control

Consists of the ACCESS and USER-ACCESS rights that are defined in the CREATE-FILE or MODIFY-FILE-ATTRIBUTES commands.

standard entity
The first *open*FT-instance that is loaded after /START-SUBSYSTEM FT. By default all *open*FT commands refer to this instance, if no other instance was specified with the command /SET-FT-INSTANCE. It is displayed as the first instance in the output of /SHOW-FT-INSTANCE INSTANCES=*ALL.

storage group
File attribute in the *virtual filestore*, encompasses the storage attributes of a file.

string
Character string

string significance
Describes the format of *strings* in files to be transferred using *FTAM protocols*.

subject
Active element in a data processing system from which an operation such as read, write, execute etc. can be initiated, that can cause a flow of information or can change the system status, e.g. ID, program, program component.

subsystem
Part of a system which processes a self-contained group of functions.

synchronous request
The user task that submitted the *FT request* waits for transfer to terminate. The user cannot continue working (see also *asynchronous request*).

SYSFILE environment
System files; the SYSFILE environment designates the totality of the system files assigned to a request.

system
-> see *FT- system*

system, local
-> see *local system*

system, remote
-> see *remote system*

system administration
– Structural unit in the computer center
– Group of individuals who employ user IDs that are associated with global privileges.

system administrator command

Command which cannot be submitted by any user ID but only by user IDs which possess the corresponding global privileges or by the TSOS user ID.

system administrator privileges

-> see *global privileges*

system files

The system input/output files assigned to a request. Users can only access system files indirectly by means of the SYSFILE command. System files provide data and resources that are required for the functions of the control program.

System files and their primary allocations:

- SYSOUT: output of system messages to terminals
- SYSLST: output of compilation logs etc.via printer (automatic SPOOLOUT)
- SYSLSTnn: as SYSLST; $1 \leq nn \leq 99$; each of the max.99 system files must be assigned to a cataloged file
- SYSOPT: output file as SYSLST
- SYSCMD: used to submit commands to the control program
- SYSDTA: used to enter data or statements

system resources

Resources in a computer system that can be requested or released by a *job* or a *task*.

task

Entity responsible for processes. In BS2000 tasks are used, among other things, to process user jobs (e.g. batch jobs, interactive jobs), see *job*.

TCP/IP (Transmission Control Protocol / Internet Protocol)

Widely used data transmission protocol (corresponds approximately to layers 3 and 4 of the *ISO/OSI reference model*, i.e. network and transport layers); originally developed for the ARPANET (computer network of the US Ministry of Defense) it has now become a de-facto standard.

TOP-SECRET

Program authored by the company Computer Associates for data and system access control.

TRANSDATA network

Data communication system that implements the TRANSDATA network concept. Products used to connect *TRANSDATA networks* to *SNA networks* include, for example, TRANSIT-CD and TRANSIT-SNA.

transfer admission

Authorization for file transfer and file management when using FTAC. The transfer admissions is then used in place of the *LOGON or LOGIN authorization*.

transfer unit

In an FTAM environment, the smallest data unit for transporting file contents. For *FTAM-1* and *FTAM-3* these are *strings*. A transfer unit can, but need not, correspond to one file record.

Transmission Control Protocol / Internet Protocol

-> see *TCP/IP*

transport connection

Logical connection between two users of the transport system (terminals or applications).

transport layer

Layer 4 of the *ISO/OSI reference model* on which the data transport protocols are handled.

transport protocol

Protocol used on the *transport layer*

transport selector (T-selector)

Subaddress used to address an ISO-8072 application in the *transport layer.*

transport system

- The part of a system or architecture that performs approximately the functions of the four lower OSI layers, i.e. the transport of messages between the two partners in a communication connection.
- Sum of the hardware and software mechanisms that allow data to be transported in computer networks.

TSN (Task Sequence Number)

Identification of a BS2000 process (*task*).

Unicode

The universal character encoding, maintained by the Unicode Consortium. This encoding standard provides the basis for processing, storage and interchange of text data in any language in all modern software and information technology protocols. The Unicode Standard defines three Unicode encoding forms: UTF-8, UTF-16 and UTF-32.

universal-class-number
Parameter of the *document-type* that defines the *character-repertoire* of a file to be transferred.

user
Represented by a *user ID*. The term “user” is a synonym for individuals, applications, procedures etc. which can obtain access to the operating system via a user ID.

user administration
-> see *global user administration*

user attribute
All the characteristics of the *user ID* that are stored in the *joinfile*.

user command
Command that can be issued under any *user identification* in system mode (/) or in program mode by means of a CMD macro.

user identification / user ID
A name with a maximum length of eight characters which is entered in the joinfile. The user ID identifies the user when accessing the system. All files and job variables are set up under a user ID. The names of the files and job variables are stored in the *file catalog* together with the user ID.

user privileges
All the attributes that represent rights that are assigned to a *user identification* and are stored in the *joinfile*.

variable length record
A record in a file all of whose records may be of different lengths. The record length must either be specified in a record length field at the start of the record or must be implicitly distinguishable from the next record through the use of a separator (e.g. Carriage Return - Line Feed).

virtual filestore
The FTAM virtual filestore is used by *FT systems* acting as *responders* to make their files available to their *partner systems*. The way a file is represented in the virtual filestore is defined in the FTAM standard, see *file attributes*.

visibleString
Character repertoire for files transferred to and from *FTAM partners*.

volume set

Component of an *SM pubset.* A volume set is a set of disks whose key properties (disk format, allocation unit) match.
The name of the volume set is administered in a directory of the SM pubset. However, the data on a volume in the volume set is addressed via the SM pubset ID.

WAN (Wide Area Network)

A public or private network that can span large distances but which runs relatively slowly and with higher error rates when compared to a *LAN*. Nowadays, however, these definitions have only limited validity. Example: in ATM networks.

Abbreviations

ACSE	Association Control Service Element
AES	Advanced Encryption Standard
ANSI	American National Standards Institute
API	Application Programming Interface
API/CS	Application Programming Interface/Communication System
APPC	Advanced Program-to-Program Communication
APPN	Advanced Peer-to-Peer Networking
ARP	Address Resolution Protocol
ASCII	American Standard Code for Information Interchange
ASECO	Advanced SEcurity COntrol (BS2000, SINIX)
ASN	Abstract Syntax Notation
ATM	Asynchronous Transfer Mode
BCAM	Basic Communication Access Method
BSI	Bundesamt für Sicherheit in der Informationstechnik
CAE	Common Application Environment
CCP	Communication Control Program
CCS	Coded Character Set
CCSN	Coded Character Set Name
CDDI	Copper Distributed Data Interface
CEN	Comité Européen de Coordination des Normes
CENELEC	Comité Européen de Normalisation Electrotechnique
CICS	Customer Information Control System (IBM)
CMX	Communication Manager SINIX
COM	Communication Port (asynchronous)
CPX	Compact Packet Exchange
DAS	Data Access Service
DAP	Directory Access Protocol

DBA	Data Base Access Service
DCAM	Data Communication Access Method
DCE	Data Communication Equipment
DCE	Distributed Computing Environment (OSF)
DCM	Data Communication Method
DDV	Datendirektverbindung (früher HfD)
DES	Data Encryption Standard (NBS)
DFR	Document File Retrieval
DFS	Distributed File System (DCE)
DIN	Deutsches Institut für Normung
DME	Distributed Management Environment
DMS	Data Management Service
DNS	Domain Name Service
DOS	Disk Operating System
DSA	Directory System Agent
DSC	Data Stream Compatibility
DSM	Distributed Systems Management
DSP	Directory System Protocol
DSS	Datensichtstation
DSSM	Dynamic Subsystem Management
DTE	Data Termination Equipment
DTS	Distributed Time Service
DVA	Datenverarbeitungsanlage
DVS	Datenverwaltungssystem
EBCDIC	Extended Binary-Coded Decimal Interchange Code
EMDS	Emulation Datensichtstation
EN	European Norm
ENV	Europäischer Normen-Vorschlag
EPHOS	European Procurement Handbook for Open Systems
ERMS	Entity Relationship Management System
ES	End System
ETSI	European Telecommunication Standards Institute
EWOS	European Workshop for Open Systems

FADU	File Access Data Unit
FDDI	Fiber Distributed Data Interface
FEP	Front End Processor
FJAM	File Job Access Method
FT	File Transfer
FTAC	File Transfer Access Control
FTAM	File Transfer, Access and Management (ISO 8571)
FTIF	File Transfer Interconnect Facility
FTP	File Transfer Protocol
GOSIP	Government OSI Profile
HDLC	High Level Data Link Control (ISO 7776)
HNC	Highspeed Net Connect
HPFS	High Performance File System
HTTP	Hypertext Transfer Protocol
IBM	International Business Machines Corporation
ICC	Intelligent Communication Controller
ICMP	Internet Control Message Protokoll
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronic Engineers
IGMP	Internet Group Management Protocol
IMS	Information Management System (IBM)
IP	Internet Protocol
ISAM	Index Sequential Access Method
ISDN	Integrated Services Digital Network
ISO	International Organization for Standardization
IT	Information Technology
ITSEC	Information Technology Security Evaluation Criteria (Europe, White Book)
ITU	International Telecommunication Union
JCL	Job Control Language
LAN	Local Area Network
LMS	Library Maintenance System
LU	Logical Unit
MAC	Medium Access Control

MAN	Metropolitan Area Network
MCR	Magnetic Card Reader
MIB	Management Information Base
MLC	Modular LAN Connect
MSV	Mittelschnelles Synchron Verfahren
MVS	Multiple Virtual System
NCP	Network Control Program (SNA)
NCS	Network Control System
NDMS	Network Data Management System
NEA	(Name der TRANSDATA-Architektur von Siemens)
NFS	Network File System
NTP	Network Time Protocol
ODI	Open Data Link Interface
ODI	Open Device Interface
ODL	Object Description Language
OSI	Open Systems Interconnection
OSS	OSI Session Service
PAM	Primary Access Method
PC	Personal Computer
PDN	Programmsystem für Datenübertragung und Netzsteuerung
PDU	Protocol Data Unit
PICS	Protocol Implementation Conformance Statement
PIN	Personal Identification Number
PLAM	Primary Library Access Method
POP	Post Office Protocol
POSIX	Portable Operating System Interface for Open Systems
PSDN	Packet Switched Data Network
PU	Physical Unit
RFC	Request for Comments
RFC1006	Request for Comments 1006
RJE	Remote Job Entry
RPC	Remote Procedure Call
RTS	Reliable Transfer Service

SAM	Sequential Access Method
SAP	Server Advertising Protocol (NetWare)
SAP	Service Access Point (OSI)
SBS	Siemens Business Services
SCM	Software Configuration Management
SDF	System Dialog Facility
SDLC	Synchronous Data Link Control
SESAM	System zur Elektronischen Speicherung Alphanumerischer Merkmale
SINIX	SNI's UNIX (SVR4)
SMTP	Simple Mail Transfer Protocol
SNA	Systems Network Architecture
SNI	Siemens Nixdorf Informationssysteme
SNMP	Simple Network Management Protocol
SQL	Structured Query Language
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol / Internet Protocol
TELNET	Telecommunications Network Protocol
TFTP	Trivial File Transfer Protocol
TID	Transport Identification
TS	Transport System
UDP	User Datagram Protocol
UDS	Universelles Datenbanksystem
URL	Uniform Resource Locator
UTM	Universal Transaction Monitor
UTM	Universeller Transaktionsmonitor
VDE	Verband deutscher Elektrotechniker
WAN	Wide Area Network
WS	Workstation
XDR	External Data Representation
XDS	API to Directory Service

Related publications

The manuals are available as online manuals, see *http://manuals.fujitsu-siemens.com*, or in printed form which must be payed and ordered separately at *http://FSC-manualshop.com*.

***open*FT for BS2000/OSD**
Enterprise File Transfer in the Open World
User Guide

***open*FT for BS2000/OSD**
Program Interfaces
Programmer Reference Guide

***open*FT for BS2000/OSD**
Messages
User Guide

***open*FT for UNIX Systems**
Enterprise File Transfer in the open world
User Guide

***open*FT for UNIX Systems**
Installation and Administration
System Administrator Guide

***open*FT for Windows**
Enterprise File Transfer in the Open World
User Guide (on-line only)

***open*FT for UNIX Systems and Windows Systems**
Program Interface
User Guide (on-line only)

***open*FT for UNIX Systems and Windows Systems**
***open*FT-Script Interface**
User Guide (on-line only)

***open*FT for z/OS**
Enterprise File Transfer in the Open World
User Guide

***open*FT for z/OS**
Installation and Administration
System Administrator Guide

***open*FTIF** (UNIX)
File Transfer Interconnect Facility in UNIX
User Guide

***open*Net Server** (BS2000/OSD)
BCAM
User Guide

SNMP Management
SNMP Management for BS2000/OSD
User Guide

BS2000/OSD-BC
Commands, Volume 1 - 5
User Guide

BS2000/OSD-BC
Commands, Volume 6, Output in S Variables and SDF-P-BASYS
User Guide

BS2000/OSD
Executive Macros
User Guide

IMON (BS2000/OSD)
Installation Monitor
User Guide

BS2000/OSD-BC
Introductory Guide to DMS
User Guide

BS2000/OSD-BC
Subsystem Management (DSSM/SSCM)
User Guide

BS2000/OSD-BC
System Installation
User Guide

BS2000/OSD-BC
Introductory Guide to Systems Support
User Guide

JV (BS2000/OSD)
Job Variables
User Guide

SECOS (BS2000/OSD
Security Control System
User Guide

XHCS (BS2000/OSD)
8-Bit Code and Unicode Support in BS2000/OSD
User Guide

RAV Computer center accounting procedures
User’s manual

HIPLEX AF (BS2000/OSD)
High availability of applications in BS2000/OSD
Product manual

Index

$FJAM 45
$SYSFJAM 18
$SYSFJAM.SYSFLF. trace file 76
$SYSFJAM.SYSLOG 55

A
abbreviated forms 83
access admission 311
access authorization 46
access check 69
access control 311
Access Control List (ACL) 311
access protection 311
access right 311
access to information 224
accounting records 305
ACL (Access Control List) 311
ACT
  explanation of output 276
action list 311
activate
  authentication check 182
  console traps 188
  FT logging function 55
  *open*FT 288
  SNMP traps 186
ACTIVE 230
  request status 230
adapt default admission set 26
add
  partner system 105
  remote system 105
ADD-FT-PARTNER 105
  enter partner system 105
address of partner system, changing 106
address of remote system, changing 106
addressing concept 44
addressing options
  internet host name 39
ADEAC
  explanation of output 276
administering
  code tables 36
  partners 38
  requests 37
administrate
  admission profiles 65
  admission set 64
administration commands, SDF 79
administrator
  FTAC 26
admission profile 65, 161, 199, 311
  administrate 65
  create (CREATE-FT-PROFILE) 127
  delete 156
  display (SHOW-FT-PROFILE) 279
  modify (example) 221
  modify (MODIFY-FT-PROFILE) 199
  modify privilege 205
  name specification 129
  privileged 65, 311, 324
  remove privilege 205
admission profile (FTAC)
  create 127
  privileged 130
admission set 169, 170, 244, 311
  administrate 64
  basic functions 131
  delete 169

admission set (cont.)
display (SHOW-FT-ADMISSION-SET) 244
modify 169
output 244
privileged 311, 312, 324
Advanced Encryption Standard (AES) 312
Advanced Security Control (ASECO) 312
AES 259
AES (Advanced Encryption Standard) 312
AET (Application Entity Title) 312
alias 88
alphanumeric 312
alphanum-name (data type) 89
ANSI code 312
API (Application Program Interface) 312
Application Entity Title (AET) 312
Application Program Interface (API) 312
ASECO (Advanced Security Control) 312
assigning a security level 106
asynchronous request 312
audit 312
personal 323
authentication 46, 312
authentication check 51
activate 182
extended 45, 268
PARTNER-CHECK 183
authorization 46
login 321
LOGON 321
authorization profiles 46

B

backup of logging records 55
basic function 171
FTAC 169
basic functions
admission set 131
limit (IGNORE-MAX-LEVELS) 131, 205
BCAM aliasing 71
BCAM processor name 39
BCMAP, MAXMAP 288
BS2000 generation for FT 16
BYTECNT
description 237
BYTE-COUNT
description 236

C

CANCEL
description 238
cancel
FT request 117
FT requests 37
CANCEL-FILE-TRANSFER 37, 117
cat (suffix for data type) 100
cat-id (data type) 89
change
address of partner system 195
CONNECTION-LIMIT 181
logging function 185
*open*FT for BS2000 public key 59
PARTNER-CHECK 183
PROCESS-LIMIT 180
security level 182
SECURITY-LEVEL 182, 195
the size of a transport unit 181
TRACE 183
TRANSPORT-UNIT-SIZE 181
changes since the last version of the manual 13
character repertoire 312
client 313
cluster 71
CNFT 117
COBOL interface 16
COBOL program 16
COBOL program interface 16
code tables 36
Comma Separated Value (CSV) 313
command overview 80
command, representation of syntax 85
command, return code 102
command-rest (data type) 89
communication computer 313
communication controller 313, 324
compatibility 17
compl (suffix for data type) 95

composed-name (data type) 89
COMPRESS, description 238
compression 313
computer network
open 313, 322
concurrency control 313
configuration user ID 18, 71, 123, 313
CONNECTION-LIMIT 34
change 181
description 181
explanation of output 268
explanation of setting 34
max. number of transport connections 181
connectivity 313
console commands 79
CONSOLE-TRAPS
description 188
explanation of output 269
constraint set 313
constructor (string) 98
contents type 314
control
diagnostic (*open*FT for BS2000) 60
locally distributed requests 106
trace function 74
corr (suffix for data type) 100, 101
create
admission profile 127
*open*FT instance 72, 123
CREATE-FT-INSTANCE 123
CREATE-FT-PROFILE 66, 127
cross-domain connection 314
c-string (data type) 89
current values of operating parameters 265

D

DATA
description 238
data 314
data communication system 314
data compression 314
data encoding 314
Data Encryption Standard (DES) 314
data protection 26, 314
data security 314
data throughput, increasing 34
data type 83
alphanum-name 89
cat-id 89
command-rest 89
composed-name 89
c-string 89
date 89
device 89
filename 90
fixed 89
integer 91
name 91
partial-name 92
posix-filename 92
posix-pathname 92
product-version 93
structured-name 93
text 93
time 93
vsn 94
x-string 94
x-text 94
data types in SDF 85, 89
suffixes 86
date (data type) 89
DDICHK 259
DEACT
explanation of output 276
deactivate
FT trace functions 183
*open*FT 291
deactivated FT requests 195
default admission set 26, 64, 170, 244, 246
adapting 26
default instance 71
default security levels 26
default value 83
define
library in admission profile 213
name of admission profile 129
selection criteria 165
transfer admission 129

delete
admission profile 156
admission set 169
FT logging records 55, 152
FT request 117
FTAC logging record 152
locked files on pubset 52
logging records 152
*open*FT instance 149
delete an instance 72
DELETE-FT-INSTANCE 149
DELETE-FT-LOGGING-RECORDS 152
DELETE-FT-PROFILE 65, 156
DENCR 259
DES 259
DES (Data Encryption Standard) 314
description
CONSOLE-TRAPS 188
FT-STATE 187
PARTNER-STATE 187
PARTNER-UNREACHABLE 187
REQUEST-QUEUE-STATE 187
SNMP-TRAPS 186
SUBSYSTEM-STATE 186
TRANSFER-FAILURE 188
TRANSFER-SUCCESS 188
device (data type) 89
diagnostic control (*open*FT for BS2000) 60
diagnostic records 77
DICHK 259
DIERR 277
DIR
description 236
directory 314
display
admission profile (SHOW-FT-PROFILE) 279
admission set 244
admission sets (example) 245
admission sets (SHOW-FT-ADMISSION-SET) 244
locked files on pubset 52
logging records 249
MAX-ADM-LEVELS 246
MAX-USER-LEVELS 246
display (cont.)
operating parameters 263
partner systems (SHOW-FT-RANGE) 285
trap controls 62
trap groups 62
trap information 63
traps 62
DNS name 39
document type 315
DSSM (Dynamic Subsystem Management) 19
dynamic partner 106
dynamic partners 43

E

ELEMENT
description 120, 229
emulation 315
enable FT logging function 19
ENCR 259
ENCRYPT 238
encryption
reject 145
restrict 145
entity 315, 319
errors, insoluble 77
evaluate *open*FT-Trace 76
example
MODIFY-FT-PROFILE 221
output fields of a logging record 256
partner system (FTAM) 113
partner system via FTIF gateway 115
partner system with FT-MFP 112
partner system with *open*FT 110
SHOW-FT-ADMISSION-SET 245
trace 74
exit *open*FT for BS2000 58
EXPANSION 137, 212
explanation
CONNECTION-LIMIT (output) 268
CONNECTION-LIMIT (setting) 34
CONSOLE-TRAPS (output) 269
LOGGING (output) 269
MAX-INBOUND-REQUEST (output) 269
MAX-REQUEST-LIFETIME (output) 269

explanation (cont.)
MAX-REQUEST-LIFETIME (setting) 35
PARTNER-CHECK (output) 268
PROCESS-LIMIT (setting) 33
REQUEST-LIMIT (output) 268
SECURITY-LEVEL (output) 268
SNMP-TRAPS (output) 269
TRACE (output) 269
TRANSPORT-UNIT-SIZE (output) 268
TRANSPORT-UNIT-SIZE (setting) 35
extended authentication check, activate 182
extended sender checking 51
extended sender checking, enable 51

F

FAILURE-PROCESSING 142, 218
FILE 239
description 120, 228, 239, 254
operand description 166, 228
selection criteria for canceling 120
file attributes 315
file catalog 315
file consistency 52
file directory 315
file management 315
file management function
modify in admission profile 219
file processing 315
file transfer
evaluate trace 76
with postprocessing 323
File Transfer Interconnect Facility 317
file transfer request 316
File Transfer, Access and Management 317
FILE-NAME
description 236
operand description 212
filename (data type) 90
FILE-PASSWORD
operand description 139, 214
filestore 315
FIN 236
FINISHED 230
request status 230
fixed (data type) 89
fixed-length record 316
FJAM-LU 316
follow-up processing 19, 140, 141, 142, 215, 216, 218, 316
follow-up processing request 316
FT administration commands, overview 80
FT administrator 316
FT administrator commands 16
FT logging 249
activate 55
enable 19
FT logging function 55
switch on 185
FT logging records 69, 249
delete 152
FT main station 44
FT requests 141, 216, 317, 326
administer 37
cancel 37, 117
deactivated, restart 195
delete 117
display max. number 268
state 230
FT setting, optimizing 31
FT system 317
start 21
stop 21
FT trace 318
FT trace function
set 183, 196
switch off 183
switch on 183
FTAC
administrator 26
admission profile (privileged) 130
basic function 169
create admission profile 127
log function 69
logging record 69
password 169, 170, 171
FTAC (File Transfer Access Control) 316
FTAC administrator 26, 316
with TSOS privilege 26, 127

FTAC functionality 316
display admission profile 279
display admission sets 244
list partner systems 285
modify admission profile 199
FTAC logging function, change 185
FTAC logging records 249
delete 152
display 249
FTAC-BS2000 316
FTAC-SINIX 316
FT-ADMINISTRATION (privilege) 16, 79
FTAM 317
FTAM file attributes 317
FTAM partner 317
addressing 39
FTAM protocol 317
FTAM-1 315
FTAM-3 315
FTIF 317
FTIF gateway 317
FTIF name 317
FT-MSP 112
FTP partner, addressing 39
FT-STATE
description 187
functional standard 318
functionality class 318

G

gateway 318
gateway processor 318
gen (suffix for data type) 100
GeneralString 312
global index 98
global privileges 318
global user administration 318
GraphicString 312
guard 318
GUARDS (Generally Usable Access Control Administration System) 319

H

heterogeneous network 319
HIPLEX cluster 71
HOLD 236
request status 230
homogeneous network 319
host 319

I

IA5String 312
IBF 245
IBP 245
IBR 245
IBS 245
IDENTIFICATION
description 196
identification 319
IDREJ 277
IGNORE-MAX-LEVELS
operand description 131, 205
inbound file management 133, 173, 208
evaluate trace 76
inbound follow-up processing 173
inbound processing 133, 208
inbound receive 132, 172, 207
inbound request 55, 319
inbound send 132, 172, 207
inbound submission 319
INBOUND-FILEMANAGEMENT 133, 173, 174, 208, 246
INBOUND-PROCESSING 133, 173, 174, 208, 246
INBOUND-RECEIVE 132, 174, 207, 246
INBOUND-SEND 132, 174, 207, 246
increased data throughput 34
index 98
INFORMATION
description 231, 255
information
about instances 72
for statistics (*open*FT for BS2000) 60
maximum number of transport connections 268
on FT requests 37, 224, 249
on FT system 54
on logging records 56

information (cont.)
on maximum number of FT requests 268
on partner systems 54
INI
description 236
initial installation 16
INITIATOR
description 227, 237, 254
operand description 135, 210
initiator 319
installation
of *open*FT for BS2000 19
of *open*FT-FTAM for BS2000 19
instance 71, 319, 322
create 72
delete 72
modify 72
select 72
set 72
instance ID 319
integer (data type) 91
integrity 53, 319
Internet 319
internet host name
addressing options 39
Internet Protocol (IP) 330, 331
interoperability 319
IPv4 address 39
IPv6 address 40
ISO reference model 320
ISO/OSI reference model 320

J
job 320
transfer 320
job class 320, 326
JBCLJOB 19
JBCLLST 19
job variable 121, 167, 229
monitoring with 167
joinfile 320
JV-PASSWORD 121, 230
operand description 167

K
kernel group 317, 320
keyed files, converting 16
keyword form 84
keyword operands 83
keywords 83

L
LAN (Local Area Network) 320
LAUTH 259, 277
length of a message 35
LIBRARY
description 120, 228
selection criteria for canceling 120
library 320
define in admission profile 213
library member 320
unrestricted access 138, 213
library name 138, 213
LIBRARY-ELEMENT 120, 228
operand description 138, 166, 213
limit basic functions (IGNORE-MAX-LEVELS) 131, 205
list partner systems 285
LOC
description 238
explanation of output 277
Local Area Network (LAN) 320
local requests, controlling 106
local system 320
LOCK 236
LOCKED 230
request status 230
log date 55
LOGGING
description 185
explanation of output 269
logging file transfer requests 55
logging function 55, 69, 320
specify 185
switch off 185
switch on 185

logging records 320
backup 55
delete 70, 152
display 249
example (long output form) 258
output 55
LOGGING-DATE
description 153
LOGGING-ID
description 251
LOGGING-TIME
description 153
Logical Unit (LU) 321
login authorization 321
LOGON authorization 133, 208, 321
long form, FT command 83
low (suffix for data type) 95
LU (logical unit) 321
LUNK
explanation of output 276

M

main station 44
mainframe 321
man (suffix for data type) 100, 101
mandatory (suffix for data type) 101
MAX-ADM-LEVEL 64, 245
description of output fields 246
maximum-string-length 321
MAX-INBOUND-REQUEST
description 185
MAX-PARTNER-LEVEL 136
MAX-REQUEST-LIFETIME 35
description 186
explanation of output 269
MAX-USER-LEVEL 171, 245
description of output fields 246
message flow control 35
metasyntax of SDF 85
MIB, *open*FT for BS2000 58
modify
admission profile 199
admission set 169
an instance 72
modify (cont.)
file management function in admission profile 219
*open*FT instance 176
operating parameters 178
partner address 194
partner characteristics 194
partner system address 196
privilege in admission profile 205
request queue 164
WRITE-MODE in admission profile 219
MODIFY-FT-ADMISSION-SET 26, 64, 169
MODIFY-FT-INSTANCE 176
MODIFY-FT-OPTIONS 32, 178
MODIFY-FT-PARTNER 37
modify partner characteristics 194
MODIFY-FT-PROFILE 65, 66, 201, 279
example 221
monitoring FT operation 55
MONJV
description 121, 229
operand description 167
selection criteria for canceling 121

N

NAME
explanation of output 276
name
change remote system 194
of the partner system 106
of the remote system 106
of the remote system, change 194
specification for admission profile 129
name (data type) 91
naming
the partner system 106
the remote system 106
naming conventions (*open*FT) 45
NCANCEL 117
cancel file transfer 117
NCP (Network Control Program) 321
NEA 321

network
heterogeneous 319
homogeneous 319
Network Control Program (NCP) 321
network description file 321
Network Management Kernel 321
NOCON, explanation of output 276
NOKEY 277
nokey files, converting 16
notational conventions for SDF 85
NSTATUS, show file transfer status 224
NUMBER, description 255
number
display file transfer requests 268
display parallel transport connections 268
display transport connections 268
of requests 34
of tasks 33, 180, 268
of transport connections 33, 34, 181
set inbound requests 185
set requests per transport connection 181
set unacknowledged send messages 181

O

object 321
OBR 245
OBS 245
odd (suffix for data type) 100
open computer network 313
*open*FT for BS2000
activate 288
diagnostic control 60
install 19
logging function 55
MIB 58
operation 29
partner 322
partner information 61
protocols 322
public key for encryption 59
start 21
start / stop 58
statistics 60
system parameter 59
*open*FT instances 71
*open*FT naming conventions 45
*open*FT partner
addressing 39
*open*FT protocol
addressing with 39
*open*FT-AC for BS2000 46, 161
security level 182
*open*FT-FTAM 322
*open*FT-FTAM for BS2000, install 19
*open*FTIF 322
operand value, constant 83
operating *open*FT 29
operating parameters 32, 322
modify 178
optimize 31
set 31
OPS variables
SHOW-FT-LOGGING-RECORDS 261
SHOW-FT-PARTNERS 274
optimizing operating parameters 31
organization of the System Administrator Guide 12
OSI reference model 320
outbound file management, evaluate trace 76
outbound receive 132, 172, 207
outbound request 55, 164, 322
outbound send 132, 172, 206
outbound submission 322
OUTBOUND-RECEIVE 132, 172, 207, 246
OUTBOUND-SEND 132, 206, 246
OUTPUT, description 264
output
admission set 244
logging records 256
overview of administration commands 80
OWNER 238
description 238
owner 117, 165, 322
of FT request 322
of the FT request 117, 119
OWNER-IDENTIFICATION 119
OWNER-IDENTIFICATION 165
description 119, 153, 227, 251

P
PACING, transport acknowledgment
behavior 181
PAMINT 16
parallel connection 181
partial-filename (data type) 92
PARTNER
description 194, 227, 236, 238, 254, 272
operand description 136, 165, 211
remove (Description) 223
partner entries
FT system active 105
in running FT system 105
partner system 323
change address 195
display (SHOW-FT-RANGE) 285
dynamic 106
list 285
via FTIF gateway (example) 115
with FT-MSP (example) 112
with *open*FT (example) 110
PARTNER-ADDRESS
description 106, 195
PARTNER-CHECK 268
authentication check 182
description 182, 268
PARTNER-NAME
description 106, 119, 272
PARTNER-STATE
description 187
PARTNER-UNREACHABLE
description 187
password 169, 323
file access, remote system 167
path-compl (suffix for data type) 95
PDN 323
permitted actions 323
Personal Audit for Individual Accountibility 323
Physical Unit (PU) 323
port number 323
partner host 40
Portable Open System Interface (POSIX) 323
positional form 84
positional operands 83
POSIX (Portable Open System Interface) 323
posix-filename (data type) 92
posix-pathname (data type) 92
postprocessing 323
preprocessing 324
preprocessor 324
presentation 324
presentation selector 324
partner host 41
print result lists 19
PRIO
description 237
PRIORITY
operand description 168
private key 324
privilege 324
PRIVILEGED 66
privileged admission profile 65, 66, 130, 324
privileged admission set 311, 312, 324
privileges, global 318
PROCESSING-ADMISSION 141, 216
operand description 139, 215
PROCESS-LIMIT
description 180
explanation of setting 33
max. number of parallel tasks 180
processor name 110
processor node 324
processor resources, optimized use 33
product-version (data type) 93
profile 325
protection during file transfer 52
protection for file transfer 51
protection mechanisms for file transfer 51
protocol 325
PRV 245
PU (Physical Unit) 323
public key for encryption 325
*open*FT for BS2000 59
public space 325
Public Volume Set 325
pubset 325
locked files display/delete 52
PW 246

Q
QUEUE-POSITION
operand description 167
quotes 83
quotes (suffix for data type) 101

R
RAUTH 259, 277
REASON-CODE
description 255
receive file 325
receive system 325
record 325
record length 316, 332
RECORD-TYPE 252
description 154, 252
REFERENCE
description 150
relay 325
relay program 326
REM
description 239
explanation of output 277
remote system 326
remove 223
remove
privileging the admission profile 205
remote system 223
REMOVE-FT-PARTNER
remove remote system 223
request 326
asynchronous 312
instance information 72
synchronous 329
request class 326
request file 326
Request for Comments (RFC) 327
request ID 326
request identification 167, 326
request information
about operating parameters 263
about partner systems 271
request management 326
request number 326
request owner 117
request queue
modify 164
request queue, modify 164
request status 230
request storage 326
REQUEST-LIMIT
explanation of output 268
REQUEST-QUEUE-STATE
description 187
requests
administering 37
controlling (locally distributed) 106
display maximum number 268
REQUEST-WAIT-LEVEL
description 181
number of requests per transport connection 181
resources 326, 330
responder 326
restart 326
restart point 327
result list 327
return code, command 102
RFC (Request for Comments) 327
RFC1006 327
Rivest-Shamir-Adleman procedure 327
router 327
RPC (Remote Procedure Call) 327
RSA 259
RSA procedure 327
RUNK
explanation of output 276

S
save logging records 152
scope ID 40
SDF, representation of syntax 85
SECLEV
explanation of output 277
SECOS 16, 79
security attributes 327
security group 317, 327
security in FT operation 46

security level 136, 169, 171, 211, 268, 327
automatic 106
change 182
for partner systems 46
SECURITY-LEVEL 46, 106
change 182, 195
description 106, 182, 195
explanation of output 268
set security level 182
SELECT
description 119, 153, 227, 251
operand description 165
selection criteria for FT requests 119
selection criteria
define 165
for FT requests to be canceled 119
for FT requests to be deleted 165
send file 328
send system 328
sep (suffix for data type) 100
server 328
service 328
service class 328
session 328
session selector 328
partner host 41
set
an instance 72
data throughput rate 34
FT trace 183
max. lifetime for inbound/outbound requests 35
max. number of inbound requests 185
max. size of transport unit 181
maximum message length 35
partner characteristics 194
security level 182
trace 196
setup
subsystem catalog entry 18
transfer admission 129
SF pubset 328
short form, FT command 83
show
FT requests 37
*open*FT for BS2000 partner information 61
SHOW-FILE-TRANSFER 37, 54
query status of FT requests 224
SHOW-FT-ADMISSION-SET 64, 244, 285
SHOW-FT-LOGGING-RECORDS 55, 56
display logging records 249
SHOW-FT-OPTIONS 54
display operating parameters 263
SHOW-FT-PARTNERS 54, 271
display partner systems 271
SHOW-FT-PROFILE 65, 66, 161, 279, 280, 281
file processing example 282
SHOW-FT-RANGE
display partner systems 285
Simple Network Management Protocol 58
Simple Network Management Protocol (SNMP) 328
Single Feature Pubset 328
size of a transport unit 268
change 181
SNA LU name 39
SNA network 328
SNMP 58
SNMP (Simple Network Management Protocol) 328
SNMP-TRAPS 62
description 186
explanation of output 269
software required 15
software requirements 15
Standard Access Control 328
standard entity 329
START
description 238
start
deactivated FT requests 195
FT system 21
*open*FT for BS2000 58
START-FT 288
startup 18

STATE
description 106, 194, 230, 236, 237
explanation of output 276
statement
representation of syntax 85
statistics
*open*FT for BS2000 60
status of FT request 230
query 224
stop
FT system 21, 291
*open*FT for BS2000 58
STOP-FT 291
storage group 317, 329
string 329
string significance 329
structured-name (data type) 93
subcode1 102
subcode2 102
subject 329
substation 44
subsystem 329
subsystem catalog entry 18
SUBSYSTEM-STATE
description 186
SUCCESS-PROCESSING 141, 216
suffixes for data types 86, 95
super FTAC administrator 26
SUSP 236
SUSPEND 230
request status 230
switch off trace 183
switch on
FT/FTAC log functions 185
trace functions 183
synchronous request 329
syntax description 85
syntax representation 85
SYSCMD 330
SYSDTA 330
SYSFILE environment 329
SYSFJAM.SYSLOG 55
log file 152
SYSFSA 71
SYSLIB.OPENFT.* 16
SYSLOG 55
SYSLST 330
SYSOPF 71
SYSOPT 330
SYSOUT 330
SYSPRG.OPENFT.* 76
SYSPTF 71
SYSRQF 71
SYSRTC.FT 16
system 329
local 320, 329
remote 326, 329
system administration 329
system administrator command 330
system administrator privileges 330
system files 330
system parameter, *open*FT for BS2000 59
system resources 330

T

task 330
Task Sequence Number (TSN) 331
TCP/IP 330, 331
temp-file (suffix for data type) 100
text (data type) 93
time (data type) 93
TOP-SECRET 330
TRACE
description 183, 196
explanation of output 277
trace
evaluate 76
set 196
switch on 183
typical examples 74
trace file $SYSFJAM.SYSFLF. 76
trace function
controlling 74
set 196
switch on 183
TRANS
description 237

TRANS-ADM
description 239
TRANSDATA network 330
transfer admission 201, 203, 279, 280, 331
define 129
transfer direction 135, 210
transfer ID 118, 165, 167, 168
transfer ID of FT request 227
transfer unit 331
TRANSFER-ADMISSION 231, 241, 245, 255, 264, 272, 281, 286
operand description 129, 202, 280
TRANSFER-DIRECTION 135, 210
TRANSFER-FAILURE
description 188
TRANSFER-ID
description 118, 227, 237
of FT request 118
operand description 165
TRANSFER-SUCCESS
description 188
TRANS-ID
description 236
TRANSIT-CD 112, 330
TRANSIT-SNA 112, 330
Transmission Control Protocol (TCP) 330, 331
TRANSP 238
transport connection 331
transport connections
define maximum number 181
display maximum number 268
transport layer 331
transport protocol 331
transport selector 331
partner host 40
transport system 331
TRANSPORT-ADDRESS
explanation of output 278
TRANSPORT-UNIT-SIZE 35, 268
change 181
description 181
explanation of output 268
explanation of setting 35
size of transport unit 181
trap controls, display 62
trap groups, display 62
trap information 63
T-selector 331
TSN (Task Sequence Number) 331
TYPE
description 121

U

under (suffix for data type) 96
universal-class-number 332
user 332
user (suffix for data type) 101
user administration 332
user attributes 332
user command 332
user ID 140, 170, 215, 244, 332
deleted 169
user identification 332
user privilege 332
USER-ADMISSION
operand description 133

V

variable-length record 332
vers (suffix for data type) 101
VERSION
description 229
version change 17
virtual filestore 332
visibleString 312, 332
volume set 333
vsn (data type) 94

W

WAIT 236
request status 230
WAN (Wide Area Network) 333
Wide Area Network (WAN) 333
wild(n) (suffix for data type) 96
wild-constr (suffix for data type) 98
with (suffix for data type) 95
with-constr (suffix for data type) 98
with-low (suffix for data type) 95

without (suffix for data type) 100
without-cat (suffix for data type) 100
without-corr (suffix for data type) 100
without-gen (suffix for data type) 100
without-man (suffix for data type) 100
without-odd (suffix for data type) 100
without-sep (suffix for data type) 100
without-user (suffix for data type) 101
without-vers (suffix for data type) 101
with-under (suffix for data type) 96
with-wild(n) (suffix for data type) 96
WRITE
    description 238
WRITE-MODE 143
    modify in admission profile 219

X

x-string (data type) 94
x-text (data type) 94

Information on this document

On April 1, 2009, Fujitsu became the sole owner of Fujitsu Siemens Computers. This new subsidiary of Fujitsu has been renamed Fujitsu Technology Solutions.

This document from the document archive refers to a product version which was released a considerable time ago or which is no longer marketed.

Please note that all company references and copyrights in this document have been legally transferred to Fujitsu Technology Solutions.

Contact and support addresses will now be offered by Fujitsu Technology Solutions and have the format *...@ts.fujitsu.com*.

The Internet pages of Fujitsu Technology Solutions are available at *http://ts.fujitsu.com/*...
and the user documentation at *http://manuals.ts.fujitsu.com*.

Copyright Fujitsu Technology Solutions, 2009

Hinweise zum vorliegenden Dokument

Zum 1. April 2009 ist Fujitsu Siemens Computers in den alleinigen Besitz von Fujitsu übergegangen. Diese neue Tochtergesellschaft von Fujitsu trägt seitdem den Namen Fujitsu Technology Solutions.

Das vorliegende Dokument aus dem Dokumentenarchiv bezieht sich auf eine bereits vor längerer Zeit freigegebene oder nicht mehr im Vertrieb befindliche Produktversion.

Bitte beachten Sie, dass alle Firmenbezüge und Copyrights im vorliegenden Dokument rechtlich auf Fujitsu Technology Solutions übergegangen sind.

Kontakt- und Supportadressen werden nun von Fujitsu Technology Solutions angeboten und haben die Form *...@ts.fujitsu.com*.

Die Internetseiten von Fujitsu Technology Solutions finden Sie unter *http://de.ts.fujitsu.com/*..., und unter *http://manuals.ts.fujitsu.com* finden Sie die Benutzerdokumentation.

Copyright Fujitsu Technology Solutions, 2009